

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE ENGENHARIA GEOGRÁFICA, GEOFÍSICA E ENERGIA



Desenvolvimento de um Portal de dados do ambiente marinho

Ana Raquel Duarte Alcântara de Melo

Mestrado em Engenharia Geográfica

Dissertação orientada por:
Prof. Doutor João Catalão
Doutor Rúben Santos

2017

UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE ENGENHARIA GEOGRÁFICA, GEOFÍSICA E ENERGIA



Desenvolvimento de um Portal de dados do ambiente marinho

Ana Raquel Duarte Alcântara de Melo

Mestrado em Engenharia Geográfica

Trabalho de dissertação orientada por:
Prof. Doutor João Catalão
Doutor Rúben Santos

Aos meus pais ...

“Para cultivar a sabedoria, é preciso força interior. Sem crescimento interno, é difícil conquistar a autoconfiança e a coragem necessárias. Sem elas, a nossa vida se complica. O impossível torna-se possível com a força de vontade.”

Dalai Lama

AGRADECIMENTOS

O propósito de concluir o mestrado, em Engenharia Geográfica, concretiza-se com a realização da presente dissertação. É o culminar de dois anos de trabalho, realizado enquanto desenvolvia a minha atividade profissional, durante os quais foi bastante importante o contributo de várias pessoas. Sem o apoio dos meus superiores, colegas, alguns familiares e amigos, não teria conseguido alcançar o objetivo traçado, pelo que, antes de passar a apresentar o trabalho de projeto que desenvolvi, gostaria de deixar umas palavras de apreço e gratidão.

Ao Prof. Doutor João Catalão, da Faculdade de Ciências da Universidade de Lisboa – DEGGE (Departamento de Engenharia Geográfica, Geofísica e Energia), e ao Doutor Rúben Santos, do Instituto Hidrográfico – divisão de Geologia Marinha, agradeço a disponibilidade permanente manifestada para assumir a orientação desta tese, pelos conhecimentos técnicos partilhados e pelas sugestões dadas durante a elaboração da mesma.

Ao Instituto Hidrográfico, na figura do seu Diretor-Geral, Contra-Almirante António M. C. Coelho Cândido, fica o meu agradecimento, por me ter permitido desenvolver o trabalho de investigação nesta instituição, nomeadamente na divisão de Geologia Marinha, onde pude contactar e tomar consciência sobre a diversidade e volume de dados usados para estudar o ambiente marinho.

O meu reconhecimento à Doutora Aurora Bizarro, chefe de divisão da Geologia Marinha do Instituto Hidrográfico, por me ter incentivado a fazer o mestrado. No decorrer destes dois anos, o seu apoio sobre o rumo a dar a este estudo, foi um contributo importante para o desenvolvimento deste trabalho.

À Administração dos Portos de Sines e do Algarve, pretendo agradecer a autorização dada para a utilização dos dados do projeto intitulado “Caraterização Ambiental da área de expansão marítima do porto de Sines e região envolvente” no desenvolvimento desta dissertação.

Quero agradecer a toda a equipa, da divisão de Geologia Marinha do Instituto Hidrográfico, em particular aos meus colegas de gabinete, André Vinhas, Sandra Moreira, Nuno Lapa e Mónica Ribeiro pelo incentivo, pelo companheirismo, pelas palavras certas nas horas de desalento, pela presença e pelas gargalhadas, ao longo deste percurso.

À minha família, em especial à minha mãe, quero agradecer todo o seu apoio e constante incentivo, paciência e compreensão demonstrados ao longo deste período que, muitas vezes, não foi um caminho fácil de percorrer, principalmente nas fases mais difíceis em que desanimei e me faltava a coragem para continuar. Ao meu irmão, fica o meu agradecimento pela ajuda e estímulo para não desistir e concluir o desafio a que me propus. Ao Diogo, meu sobrinho de cinco anos, pela compreensão manifestada, sempre que queria brincar com a tia e lhe era dito que não podia porque ela tinha muito trabalho!!

Ao Valter, pela ajuda, amizade, paciência e motivação que sempre me soube transmitir.

Aos restantes familiares e amigos que sempre me apoiaram, direta ou indiretamente, contribuindo para a superação dos obstáculos, que foram surgindo ao longo desta caminhada.

A todos os professores e colegas de mestrado, que me acompanharam ao longo do percurso académico, agradeço a forma como souberam transmitir a informação e a troca de conhecimentos proporcionada.

A todos, tendo consciência que sozinha nada disto teria sido possível, o meu muito obrigada.

RESUMO

Esta dissertação, intitulada “Desenvolvimento de um Portal de dados do ambiente marinho”, apresenta uma solução para a uniformização e disponibilização da informação deste tipo de ambiente, específica dos trabalhos realizados no Instituto Hidrográfico. Os grandes volumes de informação, com origens, datas e formatos diferentes e com distintos estádios de processamento, existentes neste instituto, restringem a sua inserção numa base de dados uniformizada, o que, consequentemente, limita o acesso da informação aos seus utilizadores. Para atingir tal objetivo, foram desenvolvidas ferramentas otimizadas à informação existente e personalizadas para futuros utilizadores, permitindo a ligação entre estes e a base de dados com a informação harmonizada do ambiente marinho. A sua concretização apoiou-se em dados marinhos adquiridos no projeto “Caraterização ambiental da área de expansão marítima do porto de Sines e região envolvente”, solicitado pela Administração do porto de Sines e coordenado pela divisão de Geologia Marinha deste instituto. Este projeto foi realizado por uma equipa multidisciplinar, que envolveu áreas técnico-científicas, tais como a geologia marinha, a química e poluição marinha, a hidrografia e a oceanografia. O presente trabalho iniciou-se com o tratamento e análise da informação correspondente a estas áreas, de modo a proceder à sua harmonização, visando integrá-la numa base de dados geográfica. Seguiu-se a sua disponibilização num portal *WebSIG*, desenvolvido neste trabalho, no qual foram criadas aplicações e disponibilizadas ferramentas, com o intuito de permitir ao utilizador a consulta dos metadados e a visualização de um conjunto de informação selecionada pelo mesmo. Na implementação da base de dados e no desenvolvimento do portal, tal como nas ferramentas, utilizou-se *software* de código aberto e gratuito, seguindo as diretivas nacionais a incentivar a utilização deste tipo de programas na Administração Pública. O sistema de gestão de base de dados relacional usado foi o *PostgreSQL*, com a extensão espacial *PostGIS*, tendo sido alimentado com a informação harmonizada do ambiente marinho. A construção de aplicações e a personalização das ferramentas foram desenvolvidas no servidor de mapas *MapServer*, sob a *framework* *p.Mapper*. Constatou-se interoperabilidade da informação geográfica nos referidos programas, o que permite concluir que há compatibilidade entre a base de dados construída em *PostgreSQL/PostGIS* e *MapServer*, *p.Mapper* e *QGIS*. Salienta-se, também, que o portal criado reúne condições para integrar qualquer tipo de informação (e.g. ortofotomapas, imagens de satélite, mapas topográficos, informação de equipamentos, elementos das equipas), inclusive de outras áreas auxiliares, nas campanhas realizadas pelo Instituto Hidrográfico.

PALAVRAS-CHAVE: *PostgreSQL/PostGIS*

MapServer

p.Mapper

Software de código aberto

Portal

ABSTRACT

This dissertation, entitled "Development of a marine environment data Portal", presents a solution for the standardisation and availability of information of this type of environment, specific to the work carried out at the Hydrographic Institute. Large volumes of information, with different origins, dates and formats and with distinct stages of processing, existing in this institute, restrict their insertion into a standardised database, which, consequently, limits the access of the information to its users. To achieve such goal, tools optimised according to the existing information and customised for future users have been developed, allowing the link between them and the database with the harmonised information of the marine environment. Its implementation was based on marine data acquired in the project "Environmental characterisation of the maritime expansion area of the port of Sines and surrounding territory", requested by the Administration of the port of Sines and coordinated by the Marine Geology division of this Institute. This project was carried out by a multidisciplinary team, which involved technical and scientific areas, such as marine geology, chemistry and marine pollution, hydrography and oceanography. The present work began with the treatment and analysis of the information corresponding to these areas, in order to harmonise it, so that it can be integrated into a geographic database. Afterwards, it was made available in a *WebSIG* portal, developed in this paper, in which applications and tools were created, to allow the user to consult the metadata and the visualisation of a set of information selected by the same. In the implementation of the database and in the development of the portal, as in the tools, was used free and open source software, following the national directives to encourage the use of this type of software in Public Administration. The relational database management system used was *PostgreSQL*, with the spatial extension *PostGIS*, and was fed with the harmonised information of the marine environment. Application building and tool customisation were developed on the *MapServer* map server under the *p.Mapper framework*. Interoperability of the geographic information in the mentioned software was verified, which allows us to conclude that there is compatibility between the database built in *PostgreSQL/PostGIS* and *MapServer*, *p.Mapper* and *QGIS*. It should also be said that the portal created combines conditions to integrate any kind of information (e.g. orthophotomaps, satellite images, topographic maps, equipment information, team elements), including other auxiliary areas, in the campaigns carried out by the Hydrographic Institute

KEYWORDS: PostgreSQL/PostGIS

MapServer

p.Mapper

Software open source

Portal

ÍNDICE

AGRADECIMENTOS.....	I
RESUMO	III
ABSTRACT	V
ÍNDICE	VII
LISTA DE FIGURAS	IX
LISTA DE TABELAS.....	XI
LISTA DE ACRÓNIMOS E ABREVIATURAS	XIII
CAPÍTULO 1 — INTRODUÇÃO	1
1.1. Enquadramento.....	1
1.2. Objetivo	2
1.3. Trabalho existente na área (relacionado).....	2
1.4. Organização do documento	5
CAPÍTULO 2 – CONCEITOS.....	7
2.1. Sistemas de Informação Geográfica.....	7
2.2. Os <i>WebSIGs</i> e a publicação de informação geográfica na internet	7
2.3. Arquitetura e componentes de um <i>WebSIG</i>	8
2.4. Gestão de Informação.....	9
2.5. Termos técnicos	10
2.5.1. Especificações OGC	10
2.5.2. Normas ISO	10
2.5.3. Formato de dados.....	11
2.5.4. Bibliotecas	11
2.6. <i>Software</i> livre e de código aberto	11
2.7. O <i>software</i> de código aberto na Administração Pública	12
2.8. <i>Software</i> de código aberto	14
2.8.1. PostgreSQL/PostGIS.....	14
2.8.2. MapServer.....	17
2.8.3. p.Mapper.....	18
2.9. Linguagens de código aberto.....	20
2.10. Aplicações utilizadas na implementação do portal	21
2.11. Metadados	22

CAPÍTULO 3 – MODELO DE DADOS	23
3.1. Origem dos Dados	23
3.2. Modelo de Dados	24
CAPÍTULO 4 – IMPLEMENTAÇÃO DO PORTAL	31
4.1. PostgreSQL/PostGIS	31
4.2. Disponibilização da informação na internet.....	38
4.2.1. MapServer.....	38
4.2.2. p.Mapper.....	42
4.2.3 Discussão	52
CAPÍTULO 5 – CONSIDERAÇÕES FINAIS	55
REFERÊNCIAS BIBLIOGRÁFICAS	57
ANEXOS	61
Anexo A — Criação das tabelas em SQL	63
Anexo B — Ficheiro <i>Mapfile</i>	67
Anexo C — <i>Scripts</i> em linguagem de programação PHP	75

LISTA DE FIGURAS

Figura 1.1	— Enquadramento da Divisão de Geologia Marinha e suas secções técnicas.	1
Figura 2.1	— Principais componentes de um <i>WebSIG</i> (Condeça, 2009).....	8
Figura 2.2	— Gestão da informação e conhecimento na AP (Vidigal, 2014).....	9
Figura 2.3	— Tipos geométricos do <i>PostgreSQL</i> (Queiroz e Ferreira, 2005).....	15
Figura 2.4	— Tipos de dados espaciais do <i>PostGIS</i> (Queiroz e Ferreira, 2005).....	16
Figura 2.5	— Arquitetura da Aplicação <i>MapServer</i> (adaptado de McKenna <i>et al.</i> , 2007).....	18
Figura 2.6	— Sistema <i>WebGIS</i> com PHP/ <i>MapScript</i> – Arquitetura (Bughi, 2007).....	19
Figura 2.7	— Conexão das camadas de informação ao <i>PostGIS</i> na aplicação <i>p.Mapper</i> (adaptado de Singh <i>et al.</i> , 2012).....	19
Figura 2.8	— Interação entre HTML, CSS e <i>JavaScript</i> (CHRIS-DURAN, 2014).....	21
Figura 2.9	— Aplicações e linguagens utilizadas.	21
Figura 3.1	— Notação gráfica para as classes do modelo OMT-G: a) Representação completa; b) Representação simplificada (adaptado de Borges, 1997; Borges <i>et al.</i> , 2001, 2005)..	25
Figura 3.2	— Meta Modelo Parcial do Modelo Geo-OMT (Borges, 1997).....	26
Figura 3.3	— Símbolos correspondentes ao <i>Geo-Objecto</i> e ao <i>Geo-Campo</i> (adaptado de Borges, 1997).....	26
Figura 3.4	— Esquema simplificado do processo de modelação de dados (ISO 19109 <i>in</i> Charneca, 2013).....	27
Figura 3.5	— Modelo de dados geográfico – diagrama de classes.	28
Figura 4.1	— Esquema para tabelas de <i>features</i> , usando SQL com tipos de geometria (adaptado de OGC, 2010).	31
Figura 4.2	— Campos da tabela: a) <i>spatial_ref_sys</i> ; b) <i>geometry_columns</i>	32
Figura 4.3	— Tipo de hierarquia geométrica em SQL (OGC, 2010).....	33
Figura 4.4	— Criação da tabela, através da importação de uma <i>shapefile</i>	34
Figura 4.5	— Constrangimentos inseridos aquando da criação da tabela.	35
Figura 4.6	— Constrangimentos validados.	36
Figura 4.7	— Tabelas criadas no: a) <i>PostgreSQL</i> ; b) <i>PostGIS</i>	36
Figura 4.8	— Expressão de uma <i>query</i> em SQL e respetivo resultado (excerto).....	37
Figura 4.9	— Função que devolve o identificador de referência espacial.	37
Figura 4.10	— Instalação: a) Pacote MS4W; b) <i>p.Mapper</i>	38
Figura 4.11	— Estrutura de um ficheiro <i>Mapfile</i>	39
Figura 4.12	— Estrutura de um <i>Mapfile</i> (Freimuth <i>et al.</i> , 2017).	40
Figura 4.13	— Exemplo da criação de uma <i>layer</i> no <i>Mapfile</i>	41
Figura 4.14	— Categorias.	43

Figura 4.15 — Organização da Interface Gráfica.	44
Figura 4.16 — Esquema de autenticação.	44
Figura 4.17 — Sistema de autenticação.	45
Figura 4.18 — Caso de uso para este protótipo.	45
Figura 4.19 — Interface gráfica com as <i>layers</i> iniciais, Linha de costa, Território emerso e o Modelo batimétrico.	46
Figura 4.20 — Lista das camadas de informação.	47
Figura 4.21 — Interface gráfica com todas as camadas de informação ativas.	47
Figura 4.22 — Mapa de referência e área de visualização: a) e b) extensão inicial do mapa; c) e d) escala de representação maior.	48
Figura 4.23 — Resultado da ferramenta <i>identify</i>	49
Figura 4.24 — Query: a) Introdução da informação; b) Excerto do resultado.	49
Figura 4.25 — <i>Plugins</i> utilizados na aplicação <i>p.Mapper</i>	50
Figura 4.26 — Transparência da <i>layer</i>	50
Figura 4.27 — Projeção utilizada na aplicação.	51
Figura 4.28 — Exemplo de aplicabilidade da ferramenta <i>locate XY</i> : a) inserção das coordenadas e b) resultado final.	51
Figura 4.29 — <i>Query</i> : a) <i>Query</i> em SQL; b) Resultados; c) Formatos para exportar a informação.	52

LISTA DE TABELAS

Tabela 1.1 — Ferramentas e funcionalidades do projeto Portal de Dados Marinhos.	5
Tabela 2.1 — Vantagens e Desvantagens do <i>Software</i> de código aberto.....	14
Tabela 2.2 — Oportunidades e Ameaças do <i>Software</i> de código aberto.....	14
Tabela 3.1 — Enquadramento técnico dos dados utilizados.	23
Tabela 3.2 — Associações segundo a ISO 19103: 2015 (adaptado de Borges <i>et al.</i> , 2001, 2005; Charneka, 2013).....	25
Tabela 3.3 — Elementos de apoio à interpretação do diagrama de classes.	29
Tabela 4.1 — Composição da barra de ferramentas.....	48

LISTA DE ACRÓNIMOS E ABREVIATURAS

ADCP — *Acoustic Doppler Current Profiler*
ANCP — Agência Nacional de Contas Públicas
ANSI C — *American National Standards Institute C*
AP — Administração Pública
APDSI — Associação para a Promoção e Desenvolvimento da Sociedade da Informação
API — *Application Programming Interface*
ASCII — *American Standard Code for Information Interchange*
BDG — Base de Dados Geográfica
CGI — *Common Gateway Interface*
CSS — *Cascading Style Sheets*
CSV — *Comma-Separated values*
DHTML — *Dynamic HTML*
DPI — *Dots per inch*
DT — Direção Técnica
EMEPC — Estrutura de Missão para a Extensão da Plataforma Continental
EMODnet — *European Marine Observation and Data Network*
EPSG — *European Petroleum Survey Group*
ESOP — Associação de Empresas de Software Open Source Portuguesas
ESRI — *Environmental Systems Research Institute*
FSF — *Free Software Foundation*
GDAL — *Geospatial Data Abstraction Library*
GeoJSON — *Geometry JavaScript Notation Object*
GIF — *Graphics Interchange Format*
GIST — *Generalized Search Trees*
GM — Geologia Marinha
GML — *Geographical Markup Language*
GNU — *GNU's Not Unix*
GPL — *General Public License*
HI — Hidrografia
HTML — *HyperText Markup Language*
HTTP — *Hypertext Transfer Protocol*
IDE — Infraestrutura de Dados Espaciais
IG — Informação Geográfica
IH — Instituto Hidrográfico
INFOMAR — *Integrated Mapping For the Sustainable Development of Ireland's MarineResource*
IPMA — Instituto Português do Mar e da Atmosfera
ISO — *International Organization for Standardization*
JPEG — *Joint Photographic Experts Group*
JSON — *JavaScript Object Notation*
KML — *Keyhole Markup Language*
LAN — *Local Area Network*
LNEG — Laboratório Nacional de Energia e Geologia
MAREANO — *Marine AREAl Database for NORwegian waters*
MDG — Modelo de Dados Geográfico
MDT — Modelo Digital do Terreno

MIT — *Massachusetts Institute of Technology*
MS4W — *MapServer For Windows*
NASA — *National American Space Agency*
OC — *Oceanografia*
OGC — *Open Geospatial Consortium*
OGR — *Simple Feature Library*
OMT — *Object Modeling Technique*
OMT-G — *Object Modeling Technique – Geographic Applications*
OpenGIS — *Open GIS Consortium*
OSGeo — *Open Source Geospatial Foundation*
OSI — *Open Source Initiative*
PDF — *Portable Document Format*
PHP — *Hypertext Preprocessor*
PNG — *Portable Network Graphics*
PROJ4 — *Cartographic Projections Library*
SFA – SQL — *Simple Feature Access – SQL option*
SFSQL — *Simple Feature Specification for SQL*
SGBD — *Sistema de Gestão de Bases de Bados*
SGBDR — *Sistema de Gestão de Bases de Bados Relacional*
SIG — *Sistemas de Informação Geográfica*
SIGD — *Sistema de Informação Geográfica Distribuído*
SNIG — *Sistema Nacional de Informação Geográfica*
SNIMar — *Sistema Nacional de Informação do Mar*
SQL — *Structured Query Language*
SQL/MM — *SQL multimedia and application packages – Part 3: Spatial*
SRID — *Spatial Reference System Identifier*
TIFF — *Tagged Image File Format*
TOC — *Table of Contents*
UML — *Unified Modeling Language*
UMN — *University of Minesota*
URL — *Uniform Resource Locator*
WAN — *Wide Area Network*
WCS — *Web Coverage Service*
WebSIG — *Sistemas de Informação Geográfica Distribuídos pela Internet*
WFS — *Web Feature Service*
WKB — *Well-know binary*
WKT — *Well-know text*
WKT-CRS — *Well-know text representation of coordinate reference systems*
WMS — *Web Map Service*
WMTS — *Web Map Tile Service*
XML — *eXtensible Markup Langu*

CAPÍTULO 1 — INTRODUÇÃO

Este capítulo encontra-se dividido em quatro secções: começa-se por fazer o enquadramento da temática em estudo, a que se segue a indicação do objetivo do projeto, a apresentação do trabalho existente na área, em que é feita uma breve descrição de alguns portais e termina com a organização do presente documento.

1.1. Enquadramento

Atualmente, vivemos numa Era Digital em que a informação é cada vez mais importante para a sociedade, ganhando extrema relevância os sistemas que permitam o arquivo, a gestão e o acesso à informação. Algumas instituições, no âmbito das suas missões, adquiriram e processaram grandes volumes de informação que, sob diversas formas e formatos, se encontra guardada e arquivada. A reutilização ou salvaguarda implica a sua conversão e harmonização (Melo *et al.*, 2016), tendo em vista, desejavelmente, a integração de toda a informação numa base de dados. Uma forma de a disponibilizar é através de uma interface *Web* (portal) de modo a estabelecer a ligação entre a base de dados e o utilizador. Há várias décadas, o Instituto Hidrográfico (IH) é um organismo que tem como missão assegurar as atividades relacionadas com as ciências e técnicas do mar (IH, 2017), o que o leva, deste modo, a ser detentor de grandes volumes de informação em ambiente marinho (Melo *et al.*, 2016).

Sendo um instituto que possui muita informação com diferentes origens, fontes e formatos, o desenvolvimento de um portal, que reunisse um conjunto de ferramentas personalizadas, para estabelecer a ligação entre o utilizador e a base de dados, com informação harmonizada deste tipo de ambiente, surgiu como uma necessidade urgente, para obviar as limitações no acesso e o comprometimento da sua gestão.

Na implementação da base de dados, foi utilizada informação de diferentes áreas técnicas do instituto (**Figura 1.1**), adquirida com o intuito de caracterizar a área de expansão marítima do porto de Sines. Este projeto, coordenado pela divisão de Geologia Marinha (GM), foi selecionado pelo seu caráter multidisciplinar e por permitir o desenvolvimento de uma base de dados alargada e abrangente.

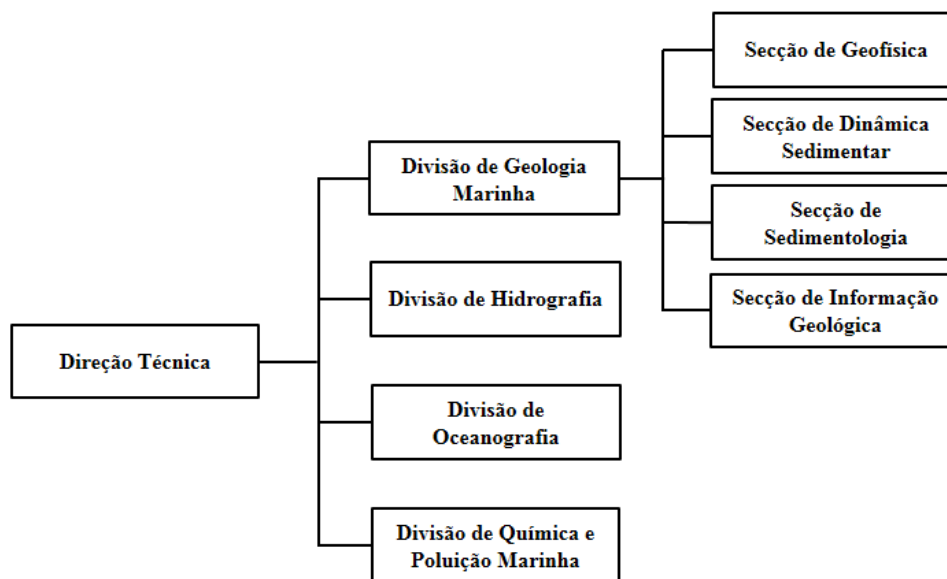


Figura 1.1 — Enquadramento da Divisão de Geologia Marinha e suas secções técnicas.

Com o propósito de enquadrar as áreas técnicas supracitadas, que são envolvidas neste estudo específico a **Figura 1.1**, espelha o organograma da direção técnica (DT) da instituição. Este mostra que a DT é composta pelas divisões de Geologia Marinha, de Química e Poluição Marinha, de Oceanografia e de Hidrografia e, por sua vez, a GM integra as secções de Geofísica, de Dinâmica Sedimentar, de Sedimentologia e de Informação Geológica. O Portal de dados marinhos irá proporcionar aos utilizadores da GM o acesso aos metadados dos vários tipos de levantamentos, que foram realizados para este projeto específico.

1.2. Objetivo

Os principais objetivos deste projeto consistem na implementação de uma base de dados geográfica (BDG) e na disponibilização de um conjunto de informação, através de um portal a desenvolver e a personalizar, em conformidade com as necessidades dos utilizadores da divisão de Geologia Marinha. Pretende-se utilizar *software* de código aberto (*open source*) e gratuito na criação da base de dados, do portal e das ferramentas. Para a concretização deste desafio, foi necessário recorrer a uma BDG conducente à conceção do portal. Esta foi criada num sistema de gestão de base de dados relacional (SGBDR) *PostgreSQL*, com a extensão espacial *PostGIS*, e com uma estrutura de classes, apresentada no modelo de dados geográfico (MDG).

Posteriormente, o portal foi concebido no servidor de mapas *MapServer*, tendo como base a *framework p.mapper*. Esta aplicação foi personalizada, tendo sido dado enfoque à inserção de ferramentas, que proporcionem aos utilizadores uma maior interação com os metadados e, por outro lado, a remoção de outras ferramentas que seriam redundantes para neste estudo. Com a finalidade de dinamizar mais esta interação, foi inserida uma ferramenta que permite a consulta dos metadados, utilizando *queries* em SQL (*Structured Query Language*).

1.3. Trabalho existente na área (relacionado)

O movimento do *software* de código aberto tem vindo a impor-se de uma forma crescente e global. Nesta linha de raciocínio, todo o procedimento efetuado como a base de dados, o portal e as ferramentas foi concebido em *software* de código aberto e gratuito. Segundo a Associação para a Promoção e Desenvolvimento da Sociedade da Informação (APDSI), este tipo de programa constitui, nos dias de hoje, uma referência nas diretivas estratégicas de um vasto número de países, regiões e organizações tanto públicas como privadas, com ou sem fins lucrativos (APDSI, 2004).

A *Free Software Foundation* (FSF) é uma organização sem fins lucrativos, que fomenta a utilização de *software livre* no quotidiano dos utilizadores. Esta constatação fundamenta-se na crescente dependência da nossa sociedade, há anos, que é bastante evidente a nível tecnológico. Em Portugal, nos últimos tempos, têm surgido iniciativas estratégicas e diretivas que visam a utilização do *software* de código aberto e gratuito, visando reduzir custos com a manutenção de licenças de programas proprietários, o que tem proporcionado uma elevada adesão a este programa devido ao apoio que se tem verificado a nível empresarial, público e individual (FSF, 2017). Em comparação com o código aberto, o custo total de propriedade apresenta uma percentagem mais elevada em soluções proprietárias, que é devido ao custo associado às atualizações ou a novas versões (APDSI, 2004), com mais impacto quando a aquisição reveste a forma de licenciamento. No que respeita à economia, o investimento do Estado é mais benéfico em *software* de código aberto do que em proprietários, dado que aqueles contribuem para a redução das importações, o que se reflete na diminuição de custo em sistemas de informação, permitindo que o remanescente recaia em investimento. Esta possível realidade traz benefícios para a economia local, favorecendo o Produto Interno Bruto.

De acordo com Bonnici (2005), a disseminação de dados espaciais e funcionalidades básicas de sistemas de informação geográfica (SIG) tornam-se financeiramente uma despesa onerosa para as instituições, motivo pelo qual esta nova realidade (o *software* livre e o de código aberto) começou a ser amplamente adotada tanto nas organizações públicas como nas privadas. O desenvolvimento surpreendente, tanto a nível científico como tecnológico, permitiu que fossem criadas aplicações especializadas com recursos para a apresentação gráfica, fazendo com que se tornasse um ambiente *Web* atraente e eficiente para a divulgação da informação geográfica (Cabral, 2008). Assim, ainda de acordo com este autor, o avanço tecnológico veio permitir que os utilizadores executem consultas à base de dados geográfica de uma forma remota.

Neste âmbito, atualmente, estão implementados vários portais sobre diversas temáticas, que disponibilizam informação em diferentes formatos. A nível nacional, na área da informação geográfica (IG), existe o geoportal do SNIG (Sistema Nacional de Informação Geográfica), que é coordenado pela Direção-Geral do Território e tem a finalidade de permitir ao utilizador efetuar a pesquisa, explorar e visualizar a IG, através dos serviços de dados geográficos da OGC (*Open Geospatial Consortium*) de diferentes tipos como o WMS (*Web Map Service*), WFS (*Web Feature Service*) e WMTS (*Web Map Tile Service*) e de ficheiros locais com diferentes formatos, como o KML (*Keyhole Markup Language*) e a *Shapefile* da ESRI (*Environmental Systems Research Institute*). Os temas que estão disponíveis para visualização são a Carta Administrativa Oficial de Portugal, a Carta de Portugal Continental à escala 1:500 000, a Geodesia do Continente, o *Corine Land Cover* 2006, 2000 e 1990 da ilha de São Miguel a Carta Geológica de Portugal e à escala 1:1 000 000. Outros temas possíveis são adicionados, através do serviço WMS, por meio do URL (*Uniform Resource Locator*), assim como de ficheiros em formato KML e *shapefile*. Este geoportal foi renovado em 2014, tendo este melhoramento incidido na pesquisa do catálogo de metadados e na introdução de outras funcionalidades, que o tornaram mais fácil de usar.

A nível de autarquias, existe o exemplo do portal de SIG do Município de Arraiolos que, em 2015, desenvolveu a aplicação na solução *MapServer*, utilizando a *framework p.Mapper*, em *software* de código aberto e gratuito. A informação geográfica, que é partilhada, incide nos limites administrativos, no cadastro geométrico, no edificado, na rede viária, hidrográfica e ferroviária, no plano diretor municipal, no plano de pormenor, entre outros.

Na temática da geologia marinha, existe o projeto internacional **OneGeology**, no site <http://onegeology.org/>, em que se refere que o mesmo surgiu da iniciativa de 81 investigadores, numa reunião realizada em Brighton (Reino Unido), em Março de 2007. A temática da reunião focou-se na viabilidade da colaboração destes membros na criação de um conjunto de mapas geológicos globais, a que os investigadores foram recetivos. O projeto é essencialmente multinacional e envolve muitas partes interessadas a trabalhar em conjunto, sendo de destacar a rede de investigação geológica em todo o mundo: a *Commission for the Geological Map of the World*, a *International Union of Geological Sciences* e a *United Nations Educational Scientific & Cultural Organisation* e o *International Framework of the International Steering Committee for Global Mapping*. Este portal disponibiliza temas como a geologia (estratigrafia, batimetria, falhas, litologia, cronoestratigrafia, entre outros), geofísica (por exemplo, gravimetria, geomagnetismo, sismologia) e a hidrologia (<http://portal.onegeology.org/OnegeologyGlobal/>). Além destes temas, existe a possibilidade de adicionar outros, através do serviço WMS (*Web Map Service*), por intermédio de um URL.

A nível europeu, em 2005, a Noruega lançou o programa **MAREANO** (*Marine AREAl Database for NORwegian waters*) que, no ano seguinte, financiou a aquisição de dados hidrográficos (multifeixe) de uma vasta área (~984 km²) em Tromsøflaket. Em 2006, o portal (http://www.mareano.no/kart/mareano_en.html?language=en) encontrava-se disponível para

visualização de mapas batimétricos, topográficos, composição de sedimentos, contaminantes, biótopos e habitats em águas norueguesas, efetuadas a partir da amostragem biológica e geológica realizadas, posteriormente, na mesma área.

Outro programa amplamente difundido, na Europa, foi o **INFOMAR** (*Integrated Mapping For the Sustainable Development of Ireland's Marine Resource*), financiado pelo Governo Irlandês, entre 2007-2013, através do seu plano de desenvolvimento. Veio suceder ao *Irish National Seabed Survey*, lançado em 2006, com o intuito de mapear as áreas costeiras do fundo do mar da Irlanda, para efeitos de caracterização ambiental, em que se incluem as baías deste país e o fundo do mar até 200m de profundidade. Abrangendo cerca de 125 000 km² das águas costeiras mais produtivas e a nível comercial mais valiosas da Irlanda, este programa fornece produtos de mapeamento integrados, que cobrem as características físicas, químicas e biológicas do fundo mar. Em 2009, foi construído o sistema interativo de disponibilização de dados na internet, tendo sido criado, em 2016, um novo visualizador de dados públicos

(<http://dcenr.maps.arcgis.com/apps/MapSeries/index.html?appid=a30af518e87a4c0ab2fbde2aaac3c228>).

Ainda a nível europeu, existe o projeto **EMODnet** (*European Marine Observation and Data Network*), que se encontra disponível, desde 2009, e que consiste num portal multidisciplinar onde se encontra um conjunto de dados de observação marinha, agrupados em portais para cada área científica (geofísica, geologia, batimetria, biologia, habitats do fundo marinho, química, física, atividades humanas e mapeamento costeiro). Sublinha-se que o portal para este projeto foi produzido com *software* de código aberto, usando a aplicação *MapServer* com a biblioteca desenvolvida em *JavaScript OpenLayers* (<http://emodnet.eu/portals>).

A nível nacional, existem também alguns portais, onde é possível aceder a alguma informação marinha e terrestre. É o caso, por exemplo, do geoportal construído, pelo Laboratório Nacional de Energia e Geologia (LNEG), em 2010, que tem nos seus horizontes a partilha de informação geoespacial, em ambiente Web, relacionada com as diferentes atividades realizadas neste laboratório, incidindo nas áreas de energia e geologia (<http://geoportal.lneg.pt/geoportal/mapas/index.html>). Os temas disponíveis focam-se em áreas técnicas como a geologia, recursos minerais, energia, recursos hidrogeológicos e recursos geotérmicos. Em relação à área da geologia, existem a Carta geológica de Portugal em várias escalas de representação, a Carta geológica da região do Algarve a 1:100 000, a Carta geológica da República da Guiné-Bissau, entre outras.

O geoportal **SNIMar** (Sistema Nacional de Informação do **Mar**) é uma infraestrutura de dados espaciais (IDE), que tem como finalidade representar a geoinformação e respetivos metadados, sobre o ambiente marinho em Portugal, partilhada pelos parceiros e pelas entidades participantes no projeto SNIMar. É um projeto que envolve parceiros dadores, Noruega, Islândia e Lichtenstein; parceiros nacionais: Estrutura de Missão para a Extensão da Plataforma Continental (EMEPC) e Instituto Português do Mar e da Atmosfera (IPMA) e oito entidades, destacando-se algumas como a Faculdade de Ciências da Universidade de Lisboa – Instituto Dom Luiz, a Direção-Geral do Território e o Instituto Hidrográfico. O geoportal (<http://geoportal.snimar.pt/#>), que foi produzido em 2017, integra temas como dados batimétricos (os recolhidos no âmbito do projeto de extensão da plataforma continental de Portugal), águas balneares de Portugal, áreas marinhas protegidas oceânicas nacionais e plataforma continental geológica na região dos Açores.

Sucintamente e visando comparar as várias ferramentas e funcionalidades existentes em cada um dos supra referidos geoportais, foi construída a **Tabela 1.1**. As ferramentas que vão ser os grandes pilares desta análise são as de navegação, interrogação, pesquisa, localização e impressão, enquanto que, por outro lado, o mapa de referência, as coordenadas do ponto e o sistema de projeção se destacam como funcionalidades. Nas ferramentas de navegação, estão integradas o *zoom in*, o *zoom out*, o *zoom full*

extent e o *pan*; nas de interrogação o *identify*; nas de pesquisa a *query*; e nas de localização o *locate XY*.

Nesta comparação, privilegiaram-se apenas estes parâmetros por terem sido considerados os principais. Contudo, é de realçar o facto de todos eles terem serviços WMS (*Web Map Service*), à exceção do programa MAREANO.

Tabela 1.1 — Ferramentas e funcionalidades do projeto Portal de Dados Marinhos.

		Projetos/Programas								
		<i>OneGeology</i>	<i>MAREANO</i>	<i>INFORMAR</i>	<i>EMODnet</i>	<i>SNIG</i>	<i>Portal SIG (Arraiolos)</i>	<i>LNEG</i>	<i>SNIMar</i>	<i>Presente estudo</i>
Ferramentas	<i>Navegação</i>	X	X	X	X	X	X	X	X	X
	<i>Interrogação</i>	X	X	X			X	X		X
	<i>Pesquisa</i>					X	X	X		X
	<i>Localização</i>		X	X			X	X		X
	<i>Impressão</i>	X	X	X		X	X	X	X	X
Funcionalidades	<i>Mapa de referência</i>	X				X	X	X		X
	<i>Coordenadas X e Y</i>	X	X	X	X	X	X	X		X
	<i>Sistema de projeção</i>	X	X		X	X	X	X		X

Analisando a tabela, verifica-se que não existem algumas ferramentas nos portais, designadamente:

- nos EMODnet, SNIG e SNIMar, a correspondente à interrogação dos elementos;
- nos OneGeology, MAREANO, INFORMAR, EMODnet e SNIMar, a de pesquisa;
- nos OneGeology, EMODnet, SNIG e SNIMar, a de localização;
- no EMODnet, a de impressão.

No que respeita às funcionalidades, constata-se que estas não são comuns a todos os portais, já que:

- não existe mapa de referência nos MAREANO, INFORMAR, EMODnet e SNIMar;
- o SNIMar não apresenta as coordenadas X e Y;
- não há indicação do sistema de projeção nos INFORMAR e SNIMar.

No caso em concreto, pretende-se disponibilizar os metadados correspondentes a informação geoespacial multidisciplinar, referentes às áreas de geologia marinha, de hidrografia e de oceanografia. A aplicação irá ser desenvolvida numa solução de código aberto e gratuito como o servidor de mapas *MapServer*, tendo como base a *framework p.Mapper* e terá as ferramentas e funcionalidades indicadas na **Tabela 1.1**.

1.4. Organização do documento

Neste documento, a descrição do desenvolvimento do Portal de dados, está estruturada em cinco capítulos, em que no:

Capítulo 1 — é exposto o enquadramento do tema da dissertação, definido o seu objetivo, apresentado um levantamento de trabalhos semelhantes, que se encontram disponíveis, cuja informação incide sobre portais diversos (nacionais, europeus e internacionais) e que fundamentam a escolha, terminando com a indicação da estrutura e organização deste relatório.

Capítulo 2 — é descrito o fundamento teórico, que enquadra as aplicações práticas.

Capítulo 3 — é apresentado o modelo de dados geográfico usado para o desenvolvimento da componente prática, com uma estrutura de classes geográfica e convencional e em conformidade com a ISO 19103: 2015.

Capítulo 4 — é descrito o procedimento efetuado na base de dados geográfica *PostgreSQL/PostGIS* e na aplicação *MapServer*, usando a *framework p.mapper*.

Capítulo 5 — são tecidas algumas considerações finais e apresentada uma prospetiva de continuidade deste projeto em trabalhos futuros.

CAPÍTULO 2 – CONCEITOS

Este capítulo inicia-se com a abordagem dos sistemas de informação geográfica, a que se seguem os *WebSIGs* e a publicação de informação geográfica na internet; na terceira parte, é feita a descrição da arquitetura e dos componentes de um *WebSIG* e, depois, referida a gestão de informação; na quinta parte, são indicados os termos técnicos a utilizar, no que respeita a especificações OGC, normas ISO, formato de dados e bibliotecas; passa-se à apresentação do *software* livre e de código aberto, que é seguido de uma alusão ao *software* de código aberto na Administração Pública; a oitava parte é dedicada ao *software* de código aberto utilizado neste projeto, especificamente, o *PostgreSQL/PostGIS*, o *MapServer* e o *p.Mapper*; as linguagens de código aberto empregadas no decorrer do trabalho estão referidas na parte nove, na décima são apresentadas as aplicações utilizadas na implementação do projeto, terminando com uma referência aos metadados.

2.1. Sistemas de Informação Geográfica

O formato *shapefile* apresenta, atualmente, vários tipos de constrangimentos que se podem traduzir em desvantagens na sua utilização, que passamos a referir. O modelo em questão, a *shapefile*, só suporta relacionamentos do tipo um para um, o que significa que cada linha da tabela de atributos estará ligada apenas a uma entidade geométrica no mapa, provocando a restrição de duas ou mais linhas não apontarem para a mesma entidade, o que se torna uma limitação para a modelação do “mundo real”. Na tabela de atributos, o nome dos campos só pode ter no máximo 10 caracteres, limitação inerente ao formato *shapefile*, e que é independente do programa que estiver a ser utilizado, *ArcGIS*, *QGIS*, *SPRING* ou outros. Este tipo de formato tem, ainda, o inconveniente de trabalhar num sistema monoutilizador o que constitui um problema, por não ser possível ter os dados armazenados em rede nem a edição da mesma *shapefile* por vários utilizadores. Para contornar este obstáculo, é necessário que cada utilizador grave uma cópia dos dados no seu computador, de modo a conseguir alterar a(s) *shapefile(s)*. No entanto, este procedimento é inadequado, por não controlar as versões mais recentes. Em contrapartida, numa base de dados geográfica, é possível trabalhar de forma eficiente com relacionamentos e número de ocorrências entre entidades, que podem ser de um para muitos (1..N) e de muitos para muitos (N..N).

Peng e Tsou (2003) referem que as novas tecnologias de informação e comunicação têm suscitado um grande impacto ao nível das ferramentas SIG, na sociedade atual. Com a sua crescente importância, principalmente ao nível da internet, os SIG *desktop* evoluíram para os SIG Distribuídos (SIGD), fazendo com que os *WebSIG* sejam considerados, hoje em dia, como potenciais ajudas ao cidadão na prestação de serviços dinâmicos e inovadores. Pode afirmar-se que os SIGD surgem em resultado da evolução normal da tecnologia, permitindo que a informação geoespacial chegue a um maior número de utilizadores e se evite a redundância de dados, tendo ainda a vantagem das funcionalidades serem similares às existentes num ambiente SIG centralizado (Longley *et al.*, 2005).

2.2. Os *WebSIGs* e a publicação de informação geográfica na internet

O termo *WebSIG* define-se como uma técnica para visualização de dados geográficos, através de aplicações *Web*, ou seja, refere-se a aplicações que disponibilizam dados espaciais aos utilizadores, servindo-se de um navegador da internet, permitindo optar-se pela internet ou intranet consoante os requisitos definidos por entidades públicas e privadas. Contudo, o facto de se partilhar informação geográfica na internet com utilizadores, que interagem com bases de dados SIG, está dependente das

capacidades do programa, já que dá a liberdade de se poder consultar, exibir e analisar remotamente a informação geoespacial por meio de uma interface de navegador da internet (*web*) (Bonnici, 2005).

No que se refere a vantagens e desvantagens na utilização do *WebSIG*, Bonnici (2005) lembra que as primeiras incidem na possibilidade de disponibilizar o acesso a dados e funcionalidades SIG a um público mais abrangente, os custos de aquisição de aplicações SIG não serem relevantes e os utilizadores não necessitarem de uma formação tão intensiva quanto um programa SIG. No respeitante a desvantagens, este autor elege como a mais importante o tempo de resposta destas aplicações, uma vez que dependem de vários fatores externos tais como: a capacidade da ligação à rede, o volume de dados transferidos, o tráfego na rede e a capacidade de processamento do servidor. É importante referir também que operações de processamento com dados pesados, normalmente, não são exequíveis em ambiente *Web* e que, embora um *WebSIG* não seja um SIG e não tenha as mesmas funções, é uma alternativa para consulta de informações que, por sua vez, estão centralizadas no servidor.

2.3. Arquitetura e componentes de um *WebSIG*

Os componentes principais, que compõem uma aplicação *WebSIG*, estão expostos no esquema da **Figura 2.1** (Condeça, 2009):

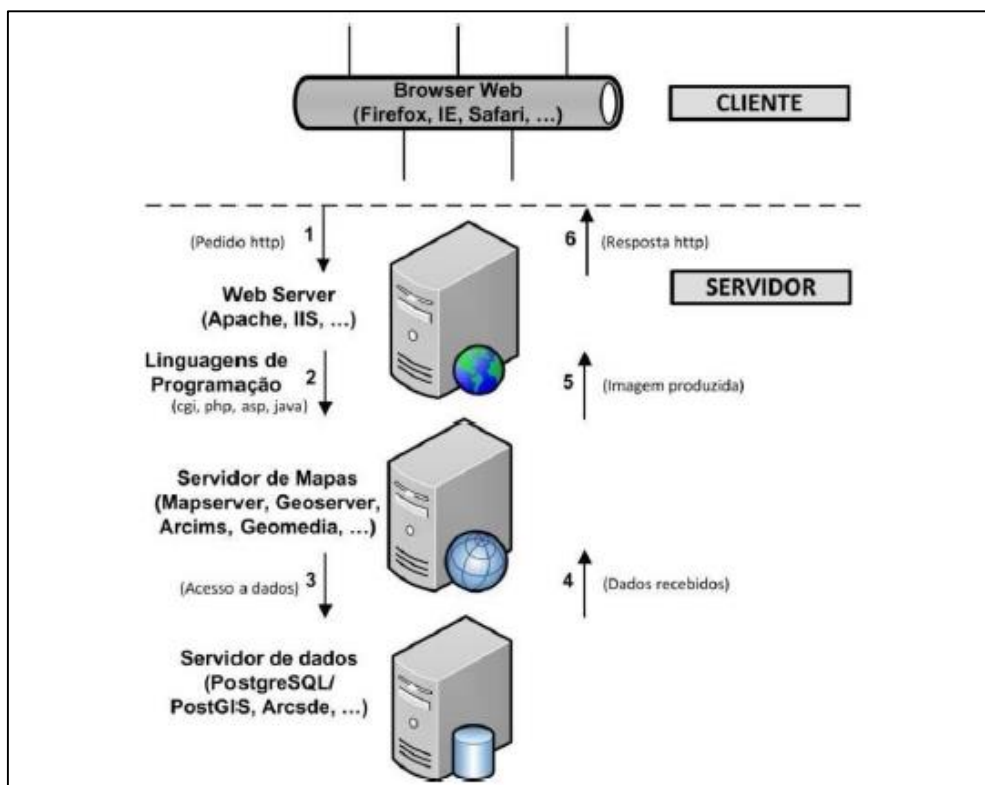


Figura 2.1 — Principais componentes de um *WebSIG* (Condeça, 2009).

- **Cliente (*Browser*)** — interface gráfica do sistema, que tem como funcionalidade permitir que o utilizador interaja com os dados espaciais e com as funções de análise fornecidas pelo *WebSIG* (Condeça, 2009). Contudo, esta interação deverá ser efetuada diretamente, através de um navegador da internet, como o *Google Chrome*, *Mozilla Firefox*, *Internet Explorer*, entre outros (Ferreira *et al.*, 2015). Salienta-se ainda que, após solicitação do utilizador, o *browser* envia o pedido URL, usando HTTP (*Hypertext Transfer Protocol*) (Catalão, 2015).
- **Servidor Web** — computador que está interligado em rede com a intenção de prover serviços para os outros *hosts*, isto é, tem como objetivo permitir o acesso a páginas, embora, na sua

essência, não tenha a função de as personalizar. Assim sendo, este tipo de servidor não interage com a base de dados, limita-se a receber as requisições do cliente (*browser*) e responder-lhe em codificação HTML (*HyperText Markup Language*) (SOFTWARELIVRE, 2017). Oliveira (2016) considera que é um programa de computador, que utiliza o HTTP como interligação entre os pedidos solicitados pelo cliente e a resposta aos mesmos com a formação da página de internet.

- **Servidor de Mapas** — aplicação que, por meio das requisições do cliente, representa os mapas através da comunicação com os dados geoespaciais (Gorni *et al.*, 2007). Para Pimenta *et al.* (2012), estes servidores permitem a interação dos utilizadores com a informação espacial que foi disponibilizada.
- **Base de dados espacial** — elemento que armazena a informação geoespacial sendo, por esta razão, a base para a construção de um GeoPortal.
- **Linguagem de Programação** — terá que ser compatível com os sistemas de informação na internet (Gorni *et al.*, 2007).

Em jeito de conclusão, Bonnici (2005) acrescenta, ainda, o componente referente ao *hardware* em que discrimina três elementos: o servidor central, o computador do cliente e a ligação via intranet ou internet, respetivamente, por meio da *Local Area Network* (LAN) ou *Wide Area Network* (WAN). Citando Catalão (2015), a LAN liga um utilizador à internet, através de *routers*, e estes, por sua vez, enviam os pedidos aos destinos na internet. No entanto, este termo está associado a redes de computadores restritas a um determinado lugar físico.

2.4. Gestão de Informação

Com a evolução da tecnologia, o fluxo da informação é proporcional ao do conhecimento. Muitas organizações, por desconhecimento dos dados que têm em seu poder, deparam-se com graves incertezas no que se refere à informação que necessitam, aquando da aquisição de novos dados. Neste âmbito, o conceito de gestão de informação é um processo que permite à organização saber usar os dados disponíveis de forma inteligente, já que, atualmente, a informação é um dos alicerces de controle de recursos na tomada de decisão. A gestão de informação consiste em compreender as necessidades de estruturação da informação, no período compreendido entre a recolha e a sua utilização. Maravilhas-Lopes (2013) considera que, no intervalo destas duas etapas, o processo passa pela identificação, classificação, processamento, armazenamento e disseminação da informação pertinente e atualizada, através da internet, em função das necessidades dos utilizadores (**Figura 2.2**).

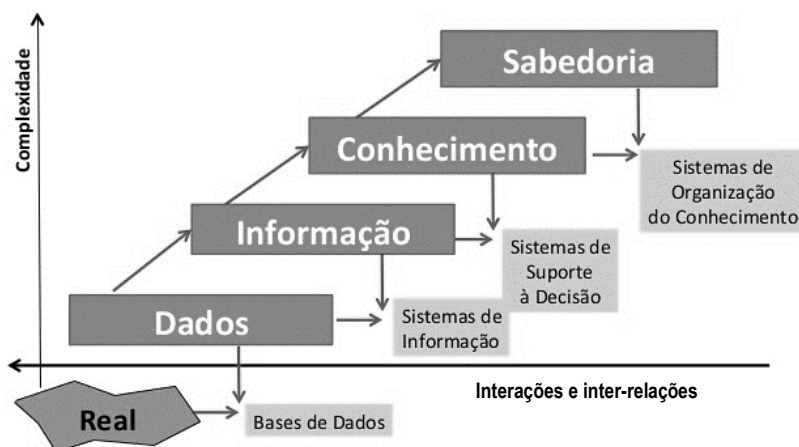


Figura 2.2 — Gestão da informação e conhecimento na AP (Vidigal, 2014).

O mesmo autor, referindo-se à relevância da gestão de informação para as organizações, realça o facto desta ser uma mais-valia, dado que traz vantagens na deteção de novas oportunidades, na criação de benefícios competitivos e, por último, proporciona a defesa de ameaças provenientes da concorrência.

2.5. Termos técnicos

Para a implementação do portal em construção, no que respeita à criação da base de dados espacial e do desenvolvimento da aplicação *Web*, neste contexto, foram tidas em consideração especificações do OGC (*Open Geospatial Consortium*), as normas ISO (*International Organization for Standardization*), alguns formatos de dados e determinadas bibliotecas, que são indicados e descritos a seguir.

2.5.1. Especificações OGC

Na década de 80, o SIG era fortemente utilizado nos domínios dos recursos naturais e defesa, especialmente dentro das agências governamentais. Contudo, os utilizadores começaram a sentir alguns constrangimentos no que concerne à limitação e ao custo da aplicação bem como à incapacidade de partilhar dados geoespaciais (OGC, 2017). Nesta linha de pensamento, em 1994, foi fundada a OGC, que é uma organização internacional sem fins lucrativos, com o compromisso de criar padrões abertos de qualidade para a comunidade geoespacial global. Esses padrões são efetuados através de um processo de consenso, entre membros de empresas de programas proprietários, entidades governamentais e universidades, e estão disponíveis gratuitamente para qualquer utilizador usar para melhorar a partilha dos dados geoespaciais do mundo (OGC, 2017). Entre os padrões disponíveis, foram considerados os seguintes:

- **GML** (*Geographical Markup Language*) — padrão de codificação desenvolvido pelo *OpenGIS* (*Open GIS Consortium*) e um formato baseado em XML para expressar características geográficas. O GML serve, por um lado, como linguagem de modelação para sistemas geográficos e, por outro, como formato de intercâmbio aberto para transações geográficas na Internet (OGC, 2017).
- **WKT** (*Well-know Text*) — linguagem de marcação de texto para representar as geometrias dos objetos em formato vetorial num mapa, sistemas de referência espacial de objetos espaciais e transformações entre estes sistemas. O WKT foi definido pela OGC e descrito nas suas especificações do *Simple Feature Access* e do Serviço de Transformação de Coordenadas. O *PostGIS* contém funções que podem converter geometrias para e de uma representação WKT, tornando-as, deste modo, legíveis para os humanos.
- **WKT-CRS** (*Well-know Text Representation of Coordinate Reference Systems*) — padrão que fornece uma versão atualizada da representação WKT de sistemas de referência de coordenadas, que segue as disposições da ISO 19111: 2007 e da ISO 1911-1: 2009 (OGC, 2017). O WKT-CRS destina-se a representar objetos em formato vetorial num mapa, no seu sistema de referência, alargando a versão anterior do WKT, para permitir a descrição das operações de coordenadas.

2.5.2. Normas ISO

- **19103: 2015** — norma que determina regras e diretrizes para o uso de uma linguagem de esquema conceitual no contexto da geoinformação. Neste enquadramento, a notação UML (*Unified Modelling Language*) é a escolhida fornecendo, no entanto, ainda o perfil para esta notação (ISO 19103, 2015).
- **SFA – SQL** (*Simple Feature Access – Part 2: SQL option*) ou **ISO 19125** — tanto é uma

especificação OGC como uma norma ISO, especificamente a 19125, que descreve um modelo comum de armazenamento e acesso de geometrias principalmente bidimensionais (ponto, linha, polígono, multiponto, multilinha, entre outros) utilizados pelos SIG.

A norma ISO 19125 está dividida em duas partes. A ISO 19125-1 (*SFA – Common Architecture* para "arquitetura comum") expõe um modelo, que é uma hierarquia de classes para características simples bidimensionais, com interpolação linear entre vértices, definindo também representação *WKT* e *WKB* (*Well-known binary*). A ISO 19125-2 (*SFA-SQL*) define uma implementação, usando SQL (OGC, 2010).

2.5.3. Formato de dados

- **XML** (*eXtensible Markup Language*) — linguagem projetada para entendimento humano, legível para máquinas e, ainda, para armazenar e transportar dados, num formato de texto simples.
- **JSON** (*JavaScript Object Notation*) — forma leve de intercâmbio de dados, acessível para os humanos lerem e escreverem (JSON, 2017).
- **GeoJSON** (*Geometry JavaScript Notation Object*) — formato de intercâmbio de dados geoespaciais baseado em *JSON*, pelo que tem a capacidade de definir vários tipos de objetos *JSON* de maneira a representar as características geográficas, as suas propriedades e extensões espaciais (Butler *et al.*, 2016). Estes autores realçam, ainda, o facto de este formato estar em conformidade com a especificação *SFSSQL*, atual *SFA – SQL* da OGC.

2.5.4. Bibliotecas

- **Proj.4** — biblioteca de rotinas referente às projeções cartográficas, de código aberto, o que permite ser acedido pelo *MapServer*, em tempo real, (Kropla, 2005). Através dela, é possível a conversão de coordenadas, geográficas e projetadas, entre os diversos sistemas de coordenadas e tipos de projeções.
- **GDAL** (*Geospatial Data Abstraction Library*) e **OGR** (*Simple Feature Library*) — bibliotecas distribuídas sob os termos da X/MIT, que é uma licença de código aberto para tradução dos dados, em formato *raster* (Kropla, 2005) e vetorial (GDAL, 2017).

2.6. Software livre e de código aberto

Em 1984, Richard Stallman funda a *Free Software Foundation* (FSF), uma entidade sem fins lucrativos, e inicia o projeto GNU (*GNU's Not Unix*), que tem como objetivo o desenvolvimento de um sistema operacional livre, ao pretender construir um programa completo que seja distribuído com o seu código-fonte (instruções que formam um programa) e em consonância com as regras de licenciamento GPL (*General Public License*) (FSF, 2017). De acordo com a definição da FSF, o *software* livre é qualquer programa de computador em que os utilizadores têm a liberdade de o usar, copiar, distribuir, estudar e melhorar sem qualquer restrição. Contudo, para um programa ser considerado *software* livre, a FSF considera que tem de conter, como base, os quatro “fundamentos” abaixo referidos (ESOP, 2017; FSF, 2017), que se traduzem na liberdade para:

- executar um programa com qualquer finalidade (liberdade 0);
- estudar o funcionamento de um programa e adaptá-lo às suas necessidades (liberdade 1), sendo o acesso ao código-fonte um pré-requisito para esta liberdade;
- redistribuir cópias do programa (liberdade 2);
- aperfeiçoar o programa e partilhar os seus melhoramentos, de modo a que toda a comunidade beneficie com essas alterações (liberdade 3), licença que também requer o acesso ao código.

Godinho (2012) reforça a ideia de que a supracitada licença GPL foi criada por Richard Stallman, em 1989, para suportar e regulamentar as 4 liberdades referidas, que caracterizam o *software* livre, bem como para as proteger, sendo atualmente a licença mais utilizada para este tipo de programa. É de ressaltar que um programa que possua uma licença GPL é de facto um *software* livre, no entanto, existem outras licenças que poderão ser utilizadas desde que sejam compatíveis com estas liberdades. Ainda neste âmbito, é de frisar que a licença GPL contempla as características do *copyleft*, termo que se foca nas permissões, o que significa que as liberdades correspondentes às modificações e às distribuições do código-fonte estão garantidas (liberdade 2 e 3).

Segundo a FSF (2017), o conceito de *software* livre advém da autonomia dada ao utilizador, ou seja, do poder que este tem de controlar a tecnologia que usa no seu computador, que se traduz em proporcionar benefícios individuais e comunitários ao invés dos programas proprietários, que possam procurar a restrição do mesmo.

Em simultâneo, existe o *software* de código aberto, criado em 1998, por iniciativa principal de Eric Raymond em colaboração com Bruce Perens e que foi desenvolvido para e por um conjunto de utilizadores, sendo distribuído sob licenças que estão de acordo com a definição de código aberto. Neste movimento, surgiu a OSI (*Open Source Initiative*) que, não ignorando as liberdades da FSF, veio estabelecer um conjunto de critérios para que um programa seja considerado de código aberto, designadamente (OPENSOURCE, 2007):

1. Redistribuição livre.
2. Disponibilização do código-fonte.
3. Autorização para se realizar alterações no código-fonte.
4. Integridade na autoria do código-fonte.
5. Não discriminação de pessoas, individualmente ou em grupo.
6. Reprovação de condicionamento a qualquer área de atividade.
7. Igualdade de direitos, associados ao programa, para distribuidores e utilizadores do mesmo.
8. Os mesmos direitos inerentes ao programa independentemente do produto.
9. Não apresentação de reservas a outros programas.
10. A licença deve ser tecnologicamente neutra.

Os 10 critérios vêm suprir a impossibilidade de integrar o código-fonte, desenvolvido em conjunto com código proprietário que, de acordo com o projeto GNU, sob a licença GPL, não são permitidos. Assim sendo, o *software* de código aberto está definido como um programa que pode ser acedido, usado, alterado e disponibilizado gratuitamente, em forma modificada ou não, por qualquer utilizador (OPENSOURCE, 2012).

2.7. O *software* de código aberto na Administração Pública

No âmbito da implementação da temática *software* livre e de código aberto, em Portugal, por ordem cronológica, destacam-se algumas iniciativas políticas e legislação nacional:

- Transposição para a legislação portuguesa de algumas das sugestões da diretiva *eEurope* 2002, prevista na **Resolução do Conselho de Ministros n.º 21/2002** (Presidência do Conselho de Ministros, Diário da República – I Série-B, n.º 26, 31 de Janeiro de 2002, fl. 886), que aponta no sentido da promoção da utilização de sistemas abertos de *software* pela Administração Pública (AP). A *eEurope* 2002 foi um plano de ação que visava estimular serviços, aplicações e conteúdos assentes numa infraestrutura de banda larga amplamente disponível, fomentando a utilização de *software* de código aberto no sector público.

- Referência direta ao *software* aberto e à sua potencial utilização na AP feita, pela primeira vez, no **Programa do XVI Governo Constitucional** (2004-2005), aprovado pelo Conselho de Ministros, em 23 de Julho de 2004.
- Recomendação ao governo da tomada de medidas com vista ao desenvolvimento do *software* livre em Portugal, conforme **Resolução da Assembleia da República n.º 66/2004**, de 23 de Setembro de 2004 (Assembleia da República, Diário da República – I Série-A, n.º 243, 15 de Outubro de 2004, fl. 6306). Das medidas propostas, no ponto 5, destaca-se o estabelecimento da obrigatoriedade de acesso ao código-fonte e especificações dos formatos de dados na aquisição de soluções informáticas destinadas à utilização pela AP.
- Decisão de utilização de normas abertas nos sistemas informáticos do Estado, definida na **Lei n.º 36/2011** (Assembleia da República, Diário da República – I Série, n.º 118, 21 de Junho de 2011, fl. 3599), na linha do Projeto de Lei n.º 421/XI/2.^a, apresentado pelo Partido Comunista Português (PCP), em 23 de Setembro de 2010. No artigo 5.º, estabelece o regulamento nacional de interoperabilidade digital.
- Deliberação sobre a adopção de normas abertas nos sistemas informáticos do Estado, definida no **Decreto-Lei n.º 98/XI**, aprovado a 6 de Abril de 2011, que teve por base o Projeto de Lei 389/XI/1.^a, proposto pelo Bloco de Esquerda (BE), em 15 de Julho de 2010 e o Projeto de Lei n.º 421/XI/2.^a, já acima referido. No p.1 do art. 4.º, impõe que “todos os processos de implementação, licenciamento ou evolução de sistemas informáticos na Administração Pública prevêm obrigatoriamente a utilização de normas abertas”.
- Recomendação de utilização de *software* aberto nos sistemas de informação do Estado, sempre que a maturidade e o custo sejam favoráveis, incluída na **Resolução do Conselho de Ministros n.º 12/2012** (Presidência do Conselho de Ministros, Diário da República – I Série, n.º 27, 7 de Fevereiro de 2012. Esta medida, surgiu como estímulo ao crescimento económico (Plano de ação, Anexo, p. 8, fl.604).

Segundo Trezentos e Simão (2004), no que concerne a diretivas internacionais, existe a *eEurope* 2005, “Uma Sociedade de Informação para Todos”, que veio suceder à diretiva *eEurope* 2002 e que se baseia em normas abertas, incentivando à utilização de *software* de fonte aberta.

Como vantagens, desvantagens, oportunidades e ameaças na utilização de *software* de código aberto, em comparação com produtos proprietários, estes autores e APDSI (2004) apontam os fatores indicados nas **Tabela 2.1** e **Tabela 2.2**.

Caçador (2008) refere que, no âmbito de um protocolo patrocinado pelo Gabinete do Plano Tecnológico, se prevê a realização de algumas iniciativas de promoção de *software* livre, salientando a vantagem que o Estado Português tem em investir no *software* de código aberto, devido ao efeito benéfico que produz na economia, uma vez que:

- reduzindo as importações de programas, a balança comercial melhora;
- sendo a aquisição de *software* de código aberto um custo que se prende com sistemas de informação, ao diminuir a sua importação, a redução do custo deixa disponível mais verba para investimento na economia local.

Nesta linha de raciocínio, é de referir que a APDSI salienta a importância do *software* de código aberto na Administração Pública, referenciando-o como um ator de peso, quando comparado com os produtos proprietários, visto criar alternativas às soluções existentes e colaborar com os princípios saudáveis da concorrência, permitindo não depender de fornecedores. A APDSI (2004) evidencia

ainda que, numa altura em que é absolutamente necessário reduzir na despesa, aquando da aquisição de soluções informáticas, deve ser tido em conta que, quer este tipo de *software* aberto quer os produtos proprietários, estão em igualdade de circunstâncias. De acordo com o estudo realizado por Godinho (2012), desde o início de 2010, tomou-se a iniciativa de incluir soluções de código aberto na lista de compras públicas do Estado Português, processo conduzido pela Agência Nacional de Contas Públicas (ANCP).

Convém ainda deixar sublinhado, neste contexto, que os produtos proprietários são os que detêm os seus direitos de uso, edição ou redistribuição. Ao não disponibilizar o seu código-fonte, os fornecedores proprietários criaram uma relação contínua de dependência com os utilizadores, formalizado pelo suporte técnico que é custeado inteiramente pelos clientes.

Tabela 2.1 — Vantagens e Desvantagens do *Software* de código aberto.

Software de código aberto	
Vantagens	Desvantagens
a flexibilidade	a possibilidade de proveniência duvidosa
o custo de licenciamento	o risco de fragmentação
a inexistência de problemas derivados do licenciamento	os problemas de instalação
a conformidade com as normas da indústria	a pobre documentação de suporte
a interoperabilidade com sistemas existentes	os problemas de segurança
a segurança	o suporte técnico escasso
o rápido desenvolvimento e correção de falhas	a ausência de abrigo legal
a capacidade de aumentar o tamanho do programa	a inexistência de pertença a uma organização
	a incerteza estratégica

Tabela 2.2 — Oportunidades e Ameaças do *Software* de código aberto.

Software de código aberto	
Oportunidades	Ameaças
a conectividade pela internet	o risco de fragmentação
os muitos distribuidores certificados	a necessidade de controlo da versão
a estrutura de suporte muito competitiva	a dependência de esforços comuns
o grande surgimento de empresas ligadas ao sector da distribuição e suporte técnico	a existência de pessoal técnico especializado
o suporte pelos “gigantes” da indústria	a forte concorrência no sector

2.8. *Software* de código aberto

2.8.1. PostgreSQL/PostGIS

PostgreSQL — é um sistema de gestão de base de dados objeto-relacional (SGBDR) resultante do pacote *POSTGRES*, que começou a ser construído em meados de 1986, numa Universidade da

Califórnia, a Universidade de Berkeley (Neto, 2003). É um projeto que foi patrocinado pela *Defense Advanced Research Projects Agency*, pelo *Army Research Office*, pela *National Science Foundation* e pela *ESL, Inc*, gratuito e de código de fonte aberto (Stonebraker *et al.*, 1990) e liderado pelo professor Michael Stonebraker.

Com base no trabalho anteriormente realizado, em 1995, Andrew Yu e Jolly Chen, sob a liderança do professor Michael Stonebraker, adicionaram um interpretador da linguagem SQL ao *POSTGRES*, o que originou o *Postgres95* cujo código foi totalmente escrito em ANSI C (*American National Standards Institute C*) (PostgreSQL, 2017). No entanto, segundo os autores Queiroz e Ferreira (2005) o código-fonte foi disponibilizado na internet (<http://www.postgresql.org>), e, desde então, os desenvolvedores vêm mantendo e aperfeiçoando aquele código sob o nome *PostgreSQL*. Segundo PostgreSQL (2017), o *PostgreSQL* é um poderoso SGBDR de código aberto, tem mais de 15 anos de desenvolvimento ativo e uma arquitetura comprovada, o que lhe valeu uma forte reputação de confiança, integridade de dados e correção.

Queiroz e Ferreira (2005) referem também que o *PostgreSQL* é dotado de tipos de dados geométricos (**Figura 2.3**), operadores espaciais simples e indexação espacial, através de uma *R-Tree* nativa, enquanto Hellerstein *et al.* (1995) consideram que é por intermédio de uma *R-Tree*, implementada no topo do mecanismo de indexação GIST (*Generalized Search Trees*). PostgreSQL (2017) acrescenta ainda que este SGBDR suporta índices compostos, únicos, parciais e funcionais que, no entanto, podem usar qualquer um dos métodos de armazenamento *B-tree*, *R-tree*, *hash* ou *GiST* (PostgreSQL, 2017). A indexação espacial é um fator importante porque tem a finalidade de fornecer mecanismos, que permitem aumentar a *performance* do sistema no momento da consulta, através de *queries*, e da recuperação dos dados espaciais (Queiroz e Ferreira, 2005).

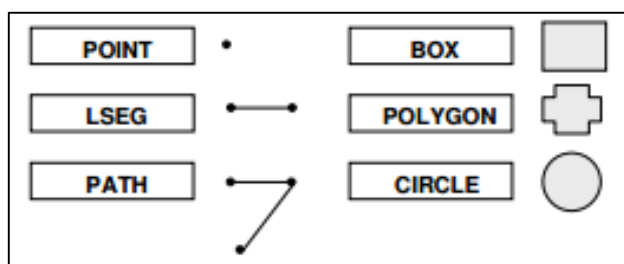


Figura 2.3 — Tipos geométricos do *PostgreSQL* (Queiroz e Ferreira, 2005).

No que diz respeito à licença, o código-fonte do *PostgreSQL* está disponível mediante uma autorização liberal de código aberto: a Licença *PostgreSQL*. Como diz PostgreSQL (2017), esta autorização permite usar, modificar e distribuir o *PostgreSQL*, de fonte aberta ou fechada, consoante o utilizador pretender. Sublinha-se que todas as alterações ou aperfeiçoamentos efetuados são do utilizador, que não tem que as partilhar ou disponibilizar. Em síntese, este SGBDR é dotado de boa capacidade de armazenamento, robusto no seu desempenho e, por último, com a aptidão de depositar e trabalhar com dados espaciais.

Akbari e Peikar (2014) apresentam um estudo comparativo entre os vários SGBD, referindo ainda que a base de dados em *MySQL* não tem implementadas todas as funções espaciais. Acrescenta, ainda, que o *PostGIS* é compatibilidade com SQL/MM, que suporta os tipos geométricos e geográficos, funções e indexação espacial.

O *PostgreSQL*, embora possua interessantes “tipos geométricos”, apresenta limitações no caso dos dados e análises em SIG, já que estes foram construídos para fins de pesquisa académica e são mais adequados para computação gráfica (PostGIS, 2017). Face ao exposto, um dos pontos fortes deste

SGBDR é o seu potencial de extensibilidade, o que possibilitou o desenvolvimento de uma extensão geográfica mais completa, o *PostGIS* (Queiroz e Ferreira, 2005).

Neste âmbito, o *PostGIS* surge com o objetivo de acrescentar a dimensão espacial ao *PostgreSQL*, tornando-o, assim, num sistema de base de dados espacial, em que são adicionados os três principais recursos espaciais: tipos, índices e funções (**Figura 2.4**). Inclui, também, no caso de tabelas de grandes dimensões, o suporte de índices espaciais (*GiST* e *R-Tree*), o que vem permitir a otimização na consulta SQL (PostGIS, 2017). Esta extensão segue as especificações do OGC “*SFA – SQL*” (anterior *OpenGIS* “*Simple Feature Specification for SQL*” (SFSQL)), atualmente certificado como compatível com o perfil “Tipos e Funções”. O desenvolvimento do *PostGIS* foi iniciado pela *Refractions Research*, em 2001, como um projeto de base de dados espacial de código aberto e foi lançado sob a licença GNU (GPLv2 ou posterior). O seu crescimento é contínuo e, desde então, têm sido acrescentados instrumentos de interface de utilizador, suporte a topologia básica, validação de dados, transformação de coordenadas, API (*Application Programming Interface*) de programação e muito mais (PostGIS, 2017). Este módulo espacial suporta os tipos de geometrias especificados pela OGC (anteriormente *OpenGIS*), apresentados na **Figura 2.4**.

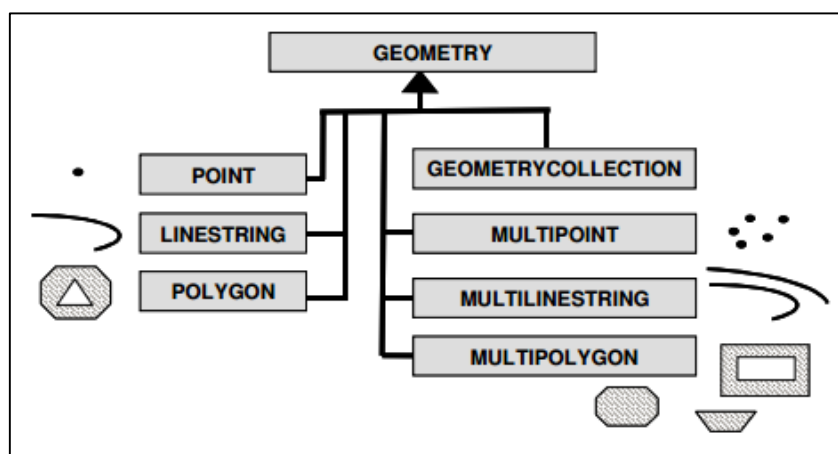


Figura 2.4 — Tipos de dados espaciais do *PostGIS* (Queiroz e Ferreira, 2005).

Uma base de dados geográfica permite que se realizem as operações de armazenamento de dados geográficos, incluindo a sua componente espacial de consultas e de análise, o que envolve os processos de edição. Neste âmbito, a BDG proporciona aos utilizadores trabalhar num sistema multiusuário, que possibilita, em simultâneo, efetuar alterações à mesma entidade geográfica ou convencional e, por outro lado, dá a possibilidade de conceder diferentes privilégios a cada utilizador. Esta última capacidade admite a existência de utilizadores com permissão apenas para visualização da informação e de outros que a podem editar, remover e atualizar. Em suma, pode dizer-se que a extensão espacial *PostGIS* confere poderes ao *PostgreSQL* para que ele trabalhe com dados geoespaciais e tenha recursos de geoprocessamento. No entanto, constata-se que o *PostgreSQL* pode ser instalado sem o *PostGIS*, salientando-se ainda o facto de que o *PostgreSQL* não trabalha com geoinformação se não tiver o *plugin PostGIS* instalado.

Como interface gráfica, este SGBDR utiliza o *pgAdmin*, que é definido como uma ferramenta de administração e gestão de código aberto para a base de dados *PostgreSQL*, servindo de apoio à manipulação da base de dados. Esta ferramenta tem a potencialidade de permitir a criação, manutenção e consulta de tabelas e a importação e exportação de dados, por meio de uma interface gráfica.

2.8.2. MapServer

MapServer — é uma solução desenvolvida em ambiente de código aberto, com a finalidade de criar aplicações de mapas interativos para a internet e é gratuito. Originalmente, em 1984, foi desenvolvido pela Universidade de Minnesota (UMN), no projeto *ForNet*, em cooperação com a *National American Space Agency* (NASA), que o subsidiou, e o Departamento de Recursos Naturais de Minnesota. Posteriormente, o *Mapserver* foi acolhido pelo projeto TerraSIP, também patrocinado pela NASA, um projeto entre o UMN e um consórcio de interesses de gestão de terras. Todavia, é uma aplicação que não pretende ser um SIG completo, sendo executado em todas as principais plataformas (*Windows*, *Linux*, *Mac OS X*) e foi lançado sob uma licença MIT (*Massachusetts Institute of Technology*) (Ingvarsson, 2004; Mapserver, 2017).

A fundação *OSGeo* (*Open Source Geospatial Foundation*) tem o *MapServer* como um dos projetos fundadores, que é mantido por programadores de todo o mundo. Um grupo diversificado de organizações apoia o financiamento do seu aperfeiçoamento e manutenção, pertencendo a administração ao *OSGeo*, através do Comité Diretor do Projeto *MapServer*, composto por aqueles especialistas e outros colaboradores. Todo o código-fonte está abertamente disponível, através do GitHub (<http://mapserver.gis.umn.edu/about.html#about>).

Esta solução, comparativamente com outros servidores de mapas existentes no mercado, é considerada por Steiniger e Hunter (2012) o servidor de mapas detentor de um excelente desempenho nas suas funcionalidades e na velocidade de processamento de conjuntos de dados de grande volume. Estes autores realçam, ainda, que o *MapServer* é considerado como sendo a ferramenta que lidera, a nível mundial, o processo de mapeamento de código aberto e que está em conformidade com as especificações da *Web WMS*, *WFS* e *WCS* (*Web Coverage Service*) da OGC. Este servidor, além de permitir a visualização dos mapas, possibilita ainda a sua interação com o utilizador. Segundo Pimenta *et al.* (2013), os servidores de mapas dinâmicos podem integrar dados geoespaciais representados por mapas temáticos, imagens de satélite, ortofotomapas e modelos 3D.

Segundo Kanegae (2005), existem três modos diferentes para o *Mapserver* ser utilizado e que passam a ser descritos:

- **CGI** (*Common Gateway Interface*) — maneira mais simples de se trabalhar com esta aplicação, em que o ficheiro executável (*mapserv.exe*) deve ser colocado numa pasta própria (*cgi-bin*) do servidor *Web Apache*. Este ficheiro recebe os parâmetros de inicialização da aplicação *webmapping*, seguidamente, processa as requisições solicitadas e, por último, reproduz ao cliente (*browser*) o mapa, a legenda, a escala, etc.
- **MapScript** — forma de combinar os recursos desta solução com a linguagem de programação que mais convier ao utilizador. Esta opção permite a criação de aplicações com um grau mais elevado de personalização (**Figura 2.5**), o que possivelmente não será conseguido, utilizando o *MapServer* em modo CGI. Em modo *MapScript*, o *MapServer* está disponível para as linguagens de programação PHP, Python, Perl, Ruby, TCL, Java e C#.
- **WebServices** — modo que o *MapServer* tem de implementar as especificações WMS, WFS e WCS do OGC, que facilitam o desenvolvimento de aplicações que o fazem funcionar como um serviço de mapas via internet. Desta forma, é possível disponibilizar os dados que serão acedidos por esta via, usando produtos *desktop* tais como: *ArcView*, *ArcExplorer*, *Jump*, *QGIS* ou *uDIG*.

De acordo com Muilwijk (2016), HTTP *Apache* foi lançado em 1995 e é o servidor *Web* mais popular e mais utilizado em cerca de 52% de todos os *websites* a nível mundial. A *Apache Software*

Foundation é uma organização sem fins lucrativos, que tem como horizontes o desenvolvimento de *software* de código aberto e, no que respeita à licença de *software* livre, o FSF considera existir compatibilidade entre as versões 2.0 da Licença *Apache* e a 3 da GPL (APACHE, 2017).

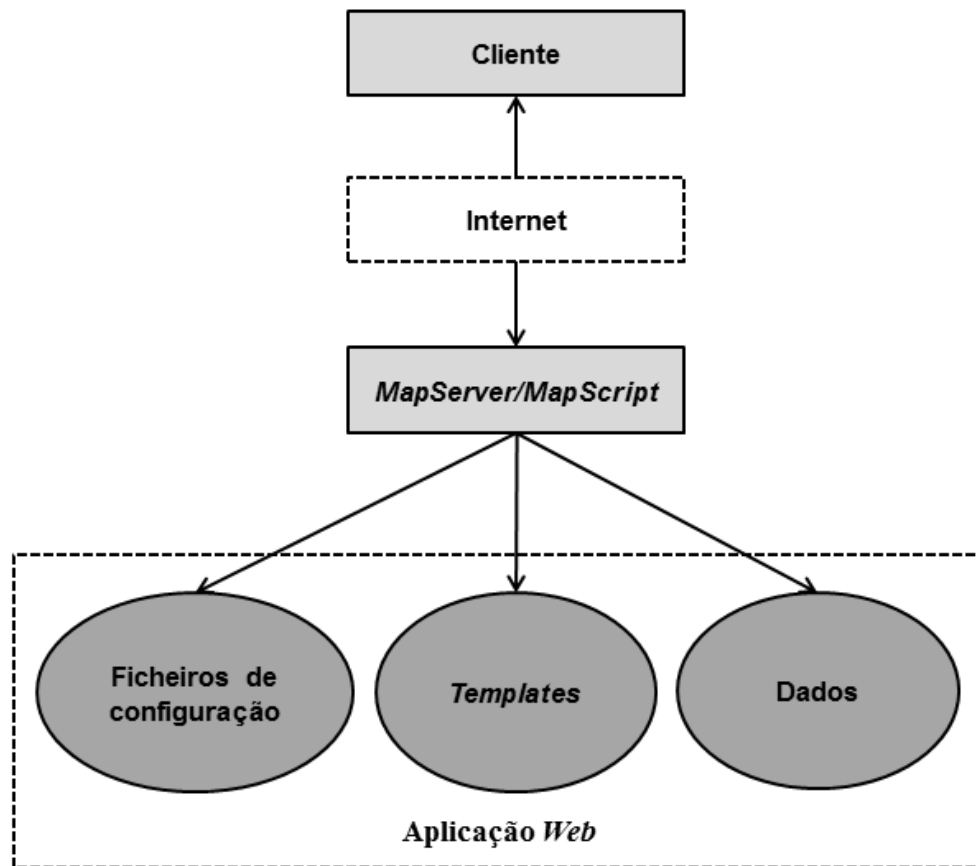


Figura 2.5 — Arquitetura da Aplicação MapServer (adaptado de McKenna *et al.*, 2007).

2.8.3. p.Mapper

p.Mapper — é uma aplicação (*framework*) desenvolvida inicialmente por Armin Burger, baseada na tecnologia do servidor de mapas *MapServer* e implementada em PHP/*MapScript* (PMAPPER, 2017), que faculta um conjunto de ferramentas prontas a serem exploradas pelo utilizador. Tem um *plugin* API para adicionar funcionalidades personalizadas e é distribuída sob a licença GNU GPL, sendo de acrescentar ainda que foi lançada a custo zero, permitindo a alteração do código-fonte e pode ser usada para qualquer fim.

Esta aplicação foi construída com diversas funcionalidades e vários *plugins*, que permitem ao utilizador criar uma aplicação *WebSIG* personalizada e interativa, ou seja, existe a possibilidade de adaptar a plataforma de acordo com os interesses e as necessidades de cada utilizador. É uma solução que admite a consulta (identificação, seleção e pesquisa) de objetos geográficos, o interface multilingue, a configuração flexível de funções, comportamento e *layout*, a impressão de mapas, a visualização da escala do mapa e da legenda, o interface DHTML (*Dynamic HTML*) (DOM) *zoom/pan* e deixa adicionar *Plugins* API. A **Figura 2.6** mostra a arquitetura de um sistema *WebSIG* com PHP/*MapScript*, em que é ilustrado o envolvimento dos diversos elementos que o integram.

O esquema seguinte (**Figura 2.7**) espelha o ciclo que a informação tem de percorrer até chegar ao utilizador de uma forma simples, acessível e intuitiva.

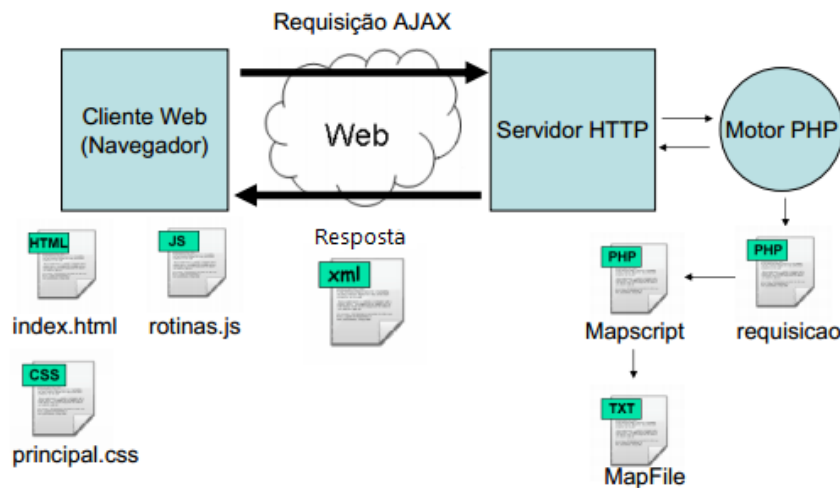


Figura 2.6 — Sistema WebGIS com PHP/MapScript – Arquitetura (Bughi, 2007).

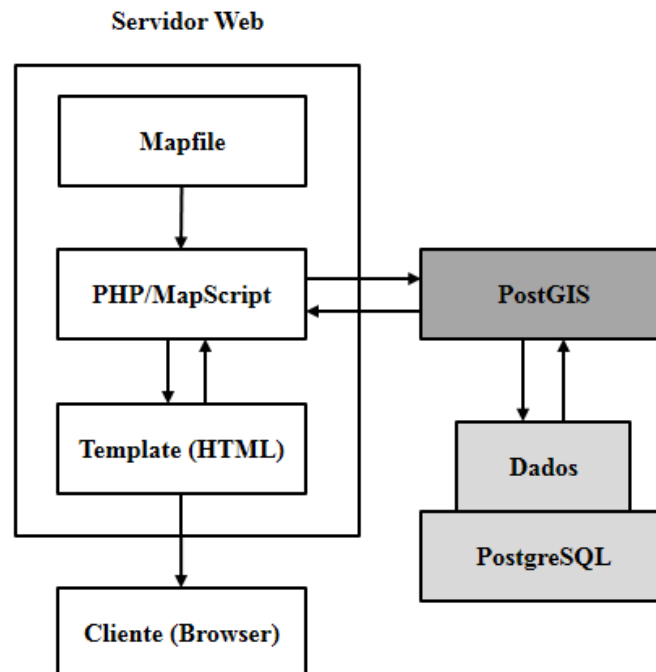


Figura 2.7 — Conexão das camadas de informação ao PostGIS na aplicação *p.Mapper* (adaptado de Singh *et al.*, 2012).

A leitura que se pode fazer desta representação é que o processo se inicia com a construção da base de dados espacial em *PostgreSQL/PostGIS*, na qual a informação é acumulada em tabelas. Como estas serão alimentadas com geoinformação, é importante frisar que é fundamental que se crie uma coluna para o campo correspondente à geometria em que cada registo recebe a sua descrição espacial. Posteriormente, estas tabelas são chamadas pelo servidor de mapas *MapServer/p.Mapper*, através do ficheiro *Mapfile*, com base nas seguintes linhas de código:

```
CONNECTIONTYPE POSTGIS
CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
DATA "geom FROM nome_tabela USING UNIQUE gid USING SRID=?"
```

Estas linhas traduzem a seguinte terminologia:

- **CONNECTIONTYPE** — tipo de ligação, isto é, nome da extensão espacial.
- **CONNECTION** — parâmetros de conexão, especificamente, o nome da base de dados que contém a tabela espacial a ser carregada, o *username*, o *host*, a *password* e a porta de configuração.
- **DATA** — nome da coluna em que está inserida a geometria, nome da tabela espacial, o campo referente à chave primária e o SRID.

Deste modo, o *MapServer* acede aos dados *PostgreSQL/PostGIS* com a potencialidade de poder exibir a informação, que consta nas tabelas da base de dados *PostgreSQL*, usufruindo do *PostGIS* como suporte de expansão espacial.

2.9. Linguagens de código aberto

HTML (*HyperText Markup Language*) — é a linguagem base dos *websites*. Não sendo o HTML um *script* executável, tem como utilidade descrever a estrutura da página de uma forma semântica. Como tal, não é considerada uma linguagem de programação, dado que é utilizada para fins estruturais e não funcionais, não seguindo uma lógica de programação. Está diretamente relacionado com a linguagem CSS (*Cascading Style Sheets*), que tem como funcionalidade a estilização, proporcionando ao programador a descrição do estilo que o documento HTML deverá exibir. São ambas linguagens que rodam na interface do cliente, ou seja, no seu navegador.

PHP (*Hypertext Preprocessor*) — é uma linguagem de *script* de código aberto (*open source*) de uso geral muito utilizada, especialmente adequada para o desenvolvimento da internet e que pode ser incluída no HTML, sendo uma linguagem de programação usada para a criação de sistemas de internet dinâmicos. Uma das características mais fortes e mais significativas do PHP é servir de suporte a uma ampla variedade de bases de dados, tendo a capacidade de abarcar informações das mesmas. É uma linguagem em que os *scripts* são interpretados no servidor e, à semelhança de outro produto CGI, permite, entre outras funções, recolher dados no formulário e gerar páginas com conteúdo dinâmico. O que distingue o PHP do *JavaScript*, no interface do cliente, é que os códigos são interpretados no servidor, gerando o HTML que é então enviado para o navegador. Este recebe os resultados da execução desse *script*, mas desconhece o código-fonte (PHP, 2017). Além do PHP, também há outras linguagens que podem rodar do lado do servidor como ASP, Java, Python e até mesmo C ou C++.

Javascript — é definida como uma linguagem de programação, usada em conjunto com o HTML, com a finalidade de permitir que as páginas escritas para a internet se tornem dinâmicas e possibilitem uma interatividade com o utilizador. Em suma, esta linguagem permite escrever pequenos *scripts* que são inseridos num documento HTML (Oliviero, 2001). O *JavaScript* é uma linguagem orientada a objetos o que irá facilitar a tarefa da programação, dado que considera todos os elementos da página como objetos distintos. À semelhança do HTML, é uma linguagem para ser executada do lado do cliente, o que é o mesmo que dizer que os *scripts* contidos nas páginas HTML são interpretados e executados, nos computadores, pelos seus utilizadores (Oliviero, 2001). O facto de ser uma linguagem interpretada significa que é de código aberto, não sendo compilada.

SQL (*Structured Query Language*) — é uma linguagem padrão para armazenar, manipular e recuperar dados numa base de dados relacional, ou seja, que visa a interação com as bases de dados fundadas no modelo relacional.

A **Figura 2.8** reflete, de forma perceptível, a interação que existe entre HTML, CSS e *Javascript*, gerando um produto final, que é matematicamente o resultado da intersecção dos três conjuntos.

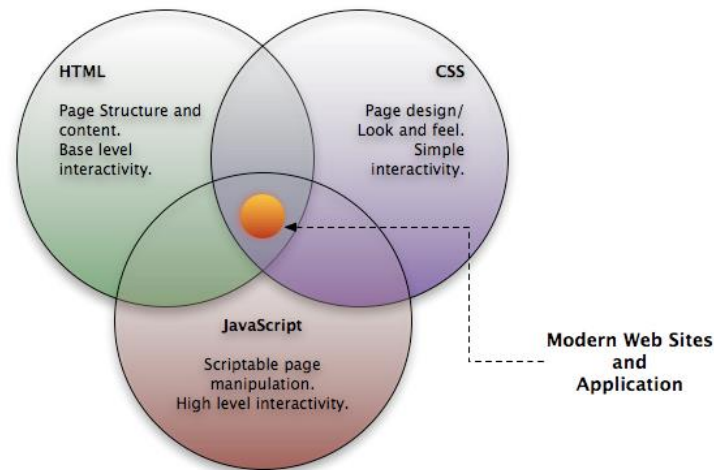


Figura 2.8 — Interação entre HTML, CSS e *JavaScript* (CHRIS-DURAN, 2014).

O ciclo de rotação deste processo inicia-se com a inserção de um URL, no *browser* (cliente), que é transmitido ao servidor; este executa uma sequência de códigos, em linguagem PHP, com o intuito de aceder à base de dados e extrair as informações necessárias para gerar o código em HTML, *JavaScript* e CSS. Por sua vez, estas serão devolvidos ao navegador (cliente) que recebe este código, executa-o e gera o produto final.

2.10. Aplicações utilizadas na implementação do portal

As soluções utilizadas, na concretização do produto final proposto na presente dissertação, estão expostas na **Figura 2.9** em que, de entre as várias aplicações de código aberto existentes e aplicáveis à AP (Trezentos e Simão, 2004), os programas foram evidenciados pelas várias etapas, de maneira a facilitar a sua interpretação.

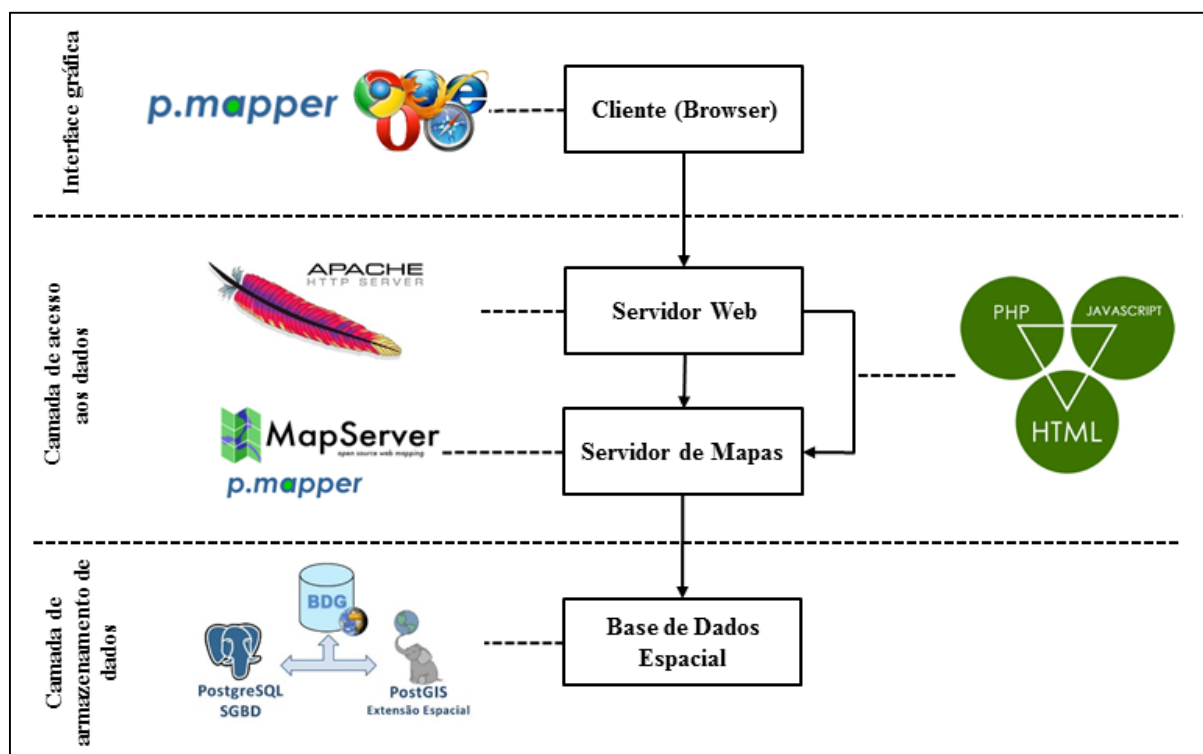


Figura 2.9 — Aplicações e linguagens utilizadas.

sendo:

- **Apache Web Server** — considerado uma referência em termos de solução de internet.
- **MapServer** — servidor de mapas.
- **p.Mapper** — *framework* de *webmapping*, baseado em *MapServer*, e desenvolvido em PHP/PHP-*Mapscript*.
- **PostGIS** — extensão espacial que permite efetuar o geoprocessamento, análise e armazenamento da informação geoespacial, vetorial ou *raster*, numa base de dados *PostgreSQL*.
- **PostgreSQL** — Sistema de Gestão de Base de Dados (SGBD) de código aberto, mais avançado que o SGBD *MySQL* em que é usado o *pgAdmin4* como interface gráfica.

2.11. Metadados

Metadados podem ser descritos como dados sobre os dados ou conjunto de dados, tendo como função principal a localização e a caracterização de uma fonte de informação. Na sua essência, os metadados têm que documentar a informação, permitem organizar, de forma coerente, os dados de acordo com certos critérios, potenciando assim o desempenho das atividades se os dados estiverem a ser utilizados adequadamente. Este facto contribui, de certa maneira, para a eficácia dos trabalhos que estiverem em curso.

Os elementos que fornecem as características importantes de um dado são, primordialmente, a fonte, o sistema de referência e a data. A fonte indica a localização geográfica do ponto onde os dados espaciais foram recolhidos e do local onde estão armazenados; o sistema de referência indica o *datum* utilizado para posicionar os dados no espaço (por exemplo, no trabalho de campo referido nas diversas campanhas); a data assinala a altura em que os dados foram adquiridos.

Esta informação bem estruturada tem o propósito de responder às seguintes questões:

- Que informação foi recolhida?
- Em que campanha os dados foram adquiridos?
- Quando é que foi obtida a informação?
- Quem é o responsável técnico da campanha?
- Qual a localização geográfica dos dados espaciais?
- Em que sistema de referência foi adquirida a informação?

CAPÍTULO 3 – MODELO DE DADOS

Este capítulo está dividido em duas secções, uma em que são apresentados os dados utilizados neste estudo, bem como uma breve descrição da sua temática e respetivo enquadramento técnico, e outra em que se mostra o modelo de dados, que representa a estrutura e organização da informação na base de dados.

3.1. Origem dos Dados

No desenvolvimento desta dissertação, foram utilizados os dados adquiridos no âmbito do projeto “Caracterização ambiental da área de expansão marítima do porto de Sines e região envolvente” solicitado ao Instituto Hidrográfico pelo porto de Sines e realizado no período de 27/01/2011 a 29/02/2012 (Rodrigues *et al.*, 2012). Na concretização deste projeto, estiveram envolvidas várias áreas técnico-científicas, nomeadamente, a da geologia marinha, a da oceanografia física e geológica, a da hidrografia e cartografia, a da geologia costeira e a da química marinha. O trabalho de campo teve que ser dividido em diversas fases, em virtude da multidisciplinaridade desejada, a par da necessidade da sua realização em situações meteorológicas distintas. Esta fase compreendeu a realização de levantamentos hidrográficos e geofísicos, campanhas de sedimentos, colheita de amostras de águas e de matéria particulada em suspensão, campanhas para fundeamento/recuperação de equipamento oceanográfico, campanhas de medição de parâmetros físicos e químicos (Rodrigues *et al.*, 2012) e monitorização costeira.

Ancorado neste enquadramento técnico, o foco principal do presente estudo incidiu nos metadados dos diferentes tipos de levantamentos apresentados. Partindo desta realidade, foi construída a **Tabela 3.1** com a finalidade de se mostrar, de forma simplificada, os dados espaciais selecionados para explorar os correspondentes metadados, fazendo sobressair o seu ambiente sedimentar e o meio envolvente.

Tabela 3.1 — Enquadramento técnico dos dados utilizados.

Áreas técnico-científicas	Dados Adquiridos	Ambiente Sedimentar
Geologia Marinha	Amostras de Águas	Plataforma Continental
		Praia
	Sedimentos Superficiais	Plataforma Continental
		Praia
	Sedimentos Verticais	Plataforma Continental
	Perfis Boomer	
Hidrografia	Sondas	Praia
Oceanografia	ADCP - fiadas	Plataforma Continental

A **Tabela 3.1** refere que, na área técnico-científica correspondente à geologia marinha, foram:

- colhidas amostras de água nos sistemas de plataforma continental e de praia, de forma a estudar o comportamento hidrodinâmico das partículas;
- adquiridas amostras de sedimentos superficiais nos mesmos sistemas, para análise das características da cobertura sedimentar do fundo e para compreensão das variações sazonais na praia de S. Torpes;
- obtidas amostras verticais da cobertura sedimentar na plataforma continental, com a finalidade de estudar a sua estrutura interna e realizar a caracterização físico-química dos sedimentos.

Estas últimas amostras foram ainda utilizadas para confirmar os dados adquiridos com os sistemas de reflexão sísmica (sistema *Boomer*), na plataforma continental, que se destinaram ao conhecimento da disposição estratigráfica e estrutural dos elementos geológicos abaixo do fundo marinho (Rodrigues *et al.*, 2012).

A área de hidrografia foi responsável pelo levantamento topo-hidrográfico, na praia de S. Torpes, com o intuito de perceber a dinâmica morfo-sedimentar dos sistemas. Por último, à oceanografia, coube estudar a variação da estrutura vertical da corrente com a maré, na área de expansão do Terminal XXI, para o que foram consideradas apenas as fiadas recolhidas com o instrumento ADCP (*Acoustic Doppler Current Profiler*) (Rodrigues *et al.*, 2012).

3.2. Modelo de Dados

O modelo de dados é um conjunto de ferramentas, que permite explicar como serão construídas as estruturas de dados, a sua organização numa base de dados e quais os relacionamentos que se pretende estabelecer entre eles. Com base nesta ideia, no presente estudo, foi construído o modelo de dados geográfico (MDG), em notação UML (*Unified Modeling Language*), no modelo de dados OMT-G (*Object Modeling Technique for Geographic Applications*). De acordo com Charneca (2013), a UML é a linguagem conceptual esquemática, que é aconselhada e adotada pelas instituições de normalização para a especificação de Informação Geográfica (IG). Por sua vez, a norma ISO 19103: 2015 fornece regras e diretrizes para o uso desta linguagem (ISO 19103, 2015), tendo definido um perfil UML para a IG.

No que respeita ao modelo OMT-G, Borges (1997) efetuou um estudo intitulado “Modelação de Dados Geográficos: Uma Extensão do Modelo OMT para Aplicações Geográficas”. Nesta investigação, realça que, entre os vários modelos de dados semânticos e orientados a objetos existentes, tais como Entidade-Relacionamento e *Object Modeling Technique* (OMT), entre outros, este último apresenta limitações para uma adequada modelação dos dados geográficos, razão pela qual apresenta o modelo OMT-G como o que mais se adapta às aplicações geográficas. Borges (1997) fundamenta este juízo, referindo que o modelo em causa tem primitivas para modelar a geometria e a topologia dos dados e oferece, como principais características, a expressividade gráfica e a capacidade de codificação (Catalão, 2015). Borges *et al.* (2001, 2005) e Catalão (2015) referem, ainda, que este modelo é baseado em três conceitos principais, nomeadamente, as classes, os relacionamentos e as restrições de integridade espaciais, sendo que os dois primeiros termos definem as primitivas básicas usadas para criar esquemas estáticos de aplicação, isto é, invariantes no tempo. Nesta linha de raciocínio e ainda segundo os mesmos autores, este modelo tem como base as primitivas definidas para o diagrama de classes da UML em que introduzem as primitivas geográficas, visando o aumento da capacidade de representação semântica daquele modelo. Neste sentido, o modelo OMT-G fornece primitivas para modelar a geometria e a topologia da informação geoespacial, no qual oferece apoio a estruturas topológicas “todo-parte” e a estruturas de rede, múltiplas representações de objetos e relacionamentos espaciais (Borges *et al.*, 2005).

A norma ISO 19103 definiu regras e recomendações para os tópicos, classes, tipos de dados, associações, atributos, estereótipos e operações (Catalão, 2015). Para uma melhor perceção desta temática, foram elaboradas a **Figura 3.1** e a **Tabela 3.2**, que refletem os vários tipos de estereótipos utilizados para este trabalho, respetivamente, os elementos gráficos da estrutura e os correspondentes às associações entre os objetos. Uma classe descreve uma coleção de objetos semelhantes, sendo um modelo em que certas características básicas de um conjunto de objetos são fixadas e define os atributos básicos e as operações dos objetos desse tipo.

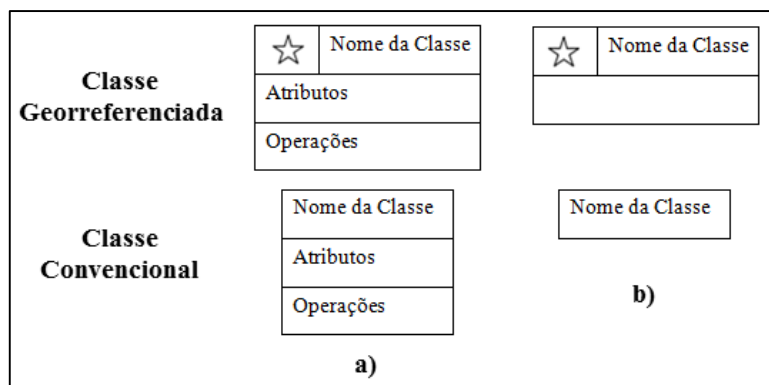


Figura 3.1 — Notação gráfica para as classes do modelo OMT-G: a) Representação completa; b) Representação simplificada (adaptado de Borges, 1997; Borges *et al.*, 2001, 2005).

A **Figura 3.1** mostra que existem dois tipos de classes, as geográficas (ou georreferenciadas) e as convencionais, e que podem ser representadas de duas formas distintas, a completa e a simplificada. A única diferença entre elas reside no facto de, na forma completa, serem discriminados os respetivos atributos. No que respeita aos dois tipos de classes, o que os distingue prende-se com a possibilidade de introduzir o tipo de geometria para as classes geográficas.

Recorda-se que definir uma classe não descreve nenhum objeto, simplesmente cria o modelo e que, para que os objetos sejam realmente gerados, as instâncias da classe devem ser desenvolvidas de acordo com a exigência do caso (**Figura 3.2**). Um objeto é caracterizado por ter uma identificação única, um conjunto de atributos, de estados e de operações (**Tabela 3.2**).

Tabela 3.2 — Associações segundo a ISO 19103: 2015 (adaptado de Borges *et al.*, 2001, 2005; Charneca, 2013).

Elementos UML para IG — Associações		
Símbolo	Nome	Descrição
	Generalização	Representa a relação entre um elemento e o subelemento que pode ser substituído pelo primeiro, isto é, uma classe é considerada a superclasse e está subdividida numa ou várias subclasses com características semelhantes.
	Associação	Representa a ligação entre dois objetos de classes diferentes, convencionais ou georreferenciadas. Esta ligação é transmitida através do grau de relação que existe entre as entidades, a destacar: <ul style="list-style-type: none"> • 0...* — pode não existir nenhum evento ou podem ocorrer vários; • 1...* — existe pelo menos uma ocorrência; • 1 — ocorre um evento; • 0...1 — não existe nenhuma ocorrência ou pode existir uma.
	Agregação	Representa uma relação parte-de (uma é o todo e a outra uma parte de), ou seja, a agregação é uma forma especial de associação entre objetos, em que um deles é considerado como sendo formado a partir de outros. Neste contexto, para as classes georreferenciadas, utiliza-se a agregação espacial em que, para estes casos, são explicitados os relacionamentos “todo-parte”. Este tipo de agregação tem duas premissas, a considerar na altura da construção do modelo, designadamente, a geometria de cada parte deve estar contida na geometria do todo e a sobreposição das partes não é permitida.

A **Tabela 3.2** mostra as notações gráficas que foram utilizadas, em linguagem UML, para representar as associações entre as classes de objetos no modelo de dados geográfico.

Borges (1997) descreve, no diagrama apresentado na **Figura 3.2**, a hierarquia de classes existente no modelo OMT-G ou Geo-OMT, no qual é de enfatizar a possível representação dos três grupos distintos encontrados nas aplicações geográficas: os contínuos, os discretos e os não espaciais.

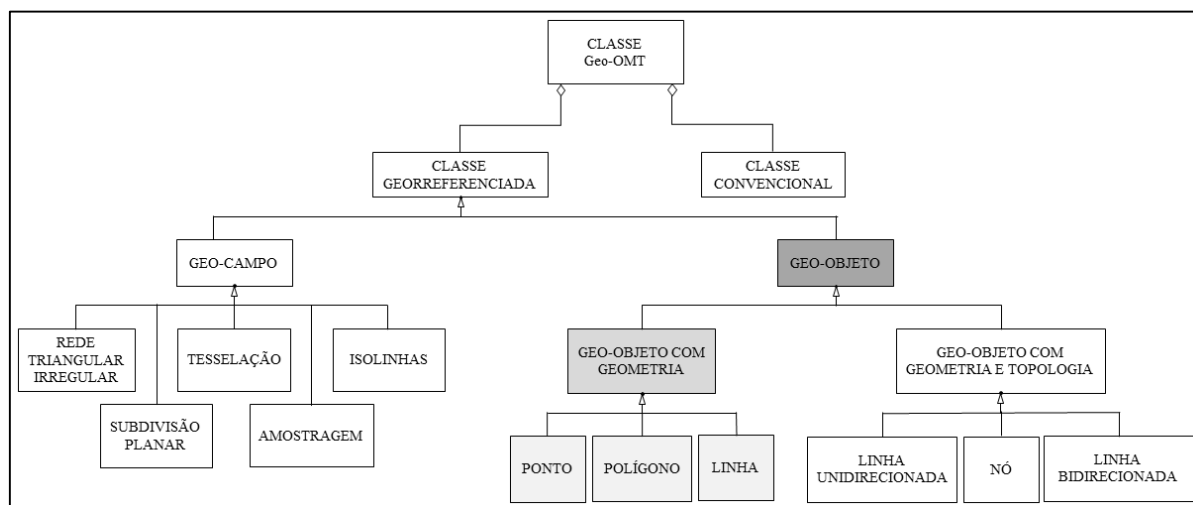


Figura 3.2 — Meta Modelo Parcial do Modelo Geo-OMT (Borges, 1997).

A figura evidencia que a classe Geo-OMT é a agregação da “classe georreferenciada” e da “classe convencional” e que as subclasses “Geo-Campo” e “Geo-Objecto” são uma generalização da superclasse “classe georreferenciada”. Câmara (2005) lembra que cada instância destas duas subclasses geográficas corresponde a uma localização no espaço e que cada uma delas tem associado um identificador único. Para este autor, estas terminologias têm o seguinte significado:

- **Geo-Campo:** um atributo que possui valores em qualquer posição no espaço geográfico.
- **Geo-Objecto:** consideram-se, como exemplo, as fronteiras que são elementos essenciais, pelo que são normalmente armazenados em estruturas poligonais, com as opções sem qualquer topologia ou arco-nó-polígono.

O presente estudo está focado na subclasse correspondente à “Coleção de Geo-Objectos” a qual, como mostra a **Figura 3.3**, está subdividida em duas entidades georreferenciadas distintas, a classe com geometria e a classe com geometria e topologia. Neste âmbito, a geoinformação a ser utilizada vai incidir apenas na classe com geometria em que a simbologia usada, para representar os tipos de geometria ponto, linha e polígono, são os que constam na referida figura.

Geo-Objecto				Geo-Campo	
Geometria		Geometria e Topologia			
—	Linha	→	Linha Uni-Direcionada		Rede Triangular Irregular
☆	Ponto	↔	Linha Bi-Direcionada		Isolinha
□	Polígono	●	Nó		Polígonos Adjacentes
					Tesselação
					Amostragem

Figura 3.3 — Símbolos correspondentes ao Geo-Objecto e ao Geo-Campo (adaptado de Borges, 1997).

A norma ISO 19109 define as várias fases a implementar na criação do modelo de dados geográfico, referentes ao Universo do Discurso, ao Modelo Conceptual, ao Modelo Lógico e aos Dados

(Charneca, 2013) (**Figura 3.4**). No entanto, como defende Borges (1997), os modelos de dados geográficos tendem a variar conforme o nível de abstração que fornecem, pelo que, para aplicações geográficas, podem ser consideradas quatro etapas distintas:

- **Universo do Discurso** (mundo real) — indica os fenómenos geográficos a representar.
- **Modelo Conceptual** (nível conceitual) — define as classes, contínuas ou discretas, que serão criadas na base de dados geográfica, originando o desenvolvimento do dicionário de dados.
- **Modelo Lógico** (nível de apresentação) — fornece as ferramentas para especificar os vários aspetos visuais diferentes, que as entidades geográficas terão que assumir quando forem utilizadas em aplicações. O catálogo de entidades é construído com base na descrição dos elementos definidos neste modelo, de acordo com a norma ISO 19110, onde estarão listadas todas as classes de entidades georreferenciadas, que foram consideradas na fase do Universo do Discurso.
- **Dados** (nível de implementação) — descreve padrões, mecanismos de armazenamento, estruturas de dados e funções para implementar fisicamente cada representação e cada apresentação consoante estiver estabelecido, respetivamente, no modelo conceptual e no modelo lógico.

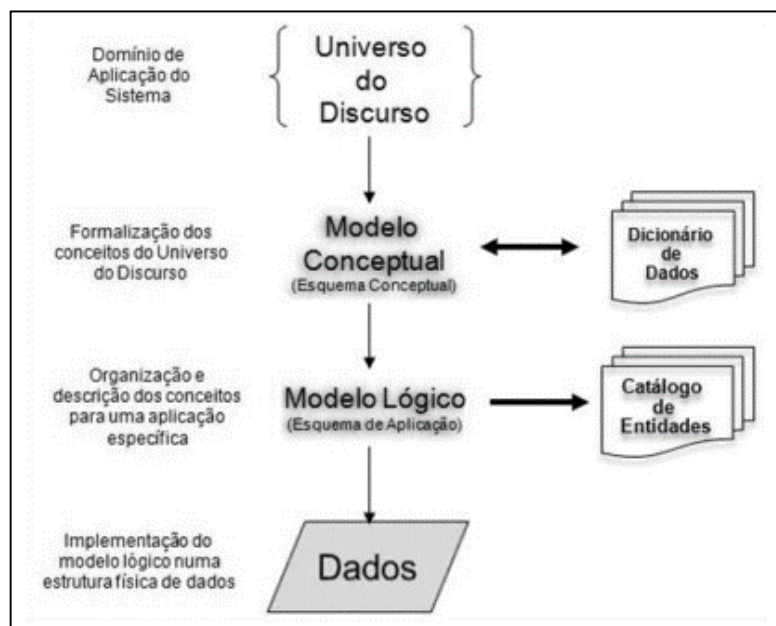


Figura 3.4 — Esquema simplificado do processo de modelação de dados (ISO 19109 in Charneca, 2013).

No desenvolvimento de uma aplicação geográfica, existem três tipos de diagramas diferentes a saber: o de classes, o de transformação e o de apresentação. Salienta-se que, no presente estudo, apenas foi construído o diagrama de classes em que foram expostas as diversas classes consideradas e os vários tipos de relacionamentos, que foram tidos em conta entre os objetos declarados.

Realça-se que este tipo de diagrama é vantajoso na medida em que facilita a visualização das relações entre as classes georreferenciadas e classes convencionais, o que propicia que a construção da base de dados geográfica seja mais intuitiva. Não nos podemos esquecer que Borges *et al.* (2001, 2005) advertem que, a partir deste diagrama, é possível formar-se um conjunto de restrições de integridade espaciais (aspetos topológicos e geométricos dos dados), a ter em atenção no nível de implementação, de modo a preservar a semântica e a qualidade dos dados armazenados.

O modelo de dados geográfico, elaborado para este trabalho, está exposto na **Figura 3.5** por meio de um diagrama de classes de acordo com Borges (1997) e Borges *et al.* (2001, 2005), no qual, para cada

entidade geográfica e convencional, foram inseridos os respetivos metadados adequadamente retirados do relatório elaborado por Rodrigues *et al.* (2012). O relatório inclui um SIG composto pelas *shapefiles* das várias áreas técnico-científicas já referidas na **Tabela 3.1**, em que cada tabela de atributos foi alimentada com informação considerada relevante para o projeto mencionado. Para tal, na informação que foi extraída destas tabelas e do relatório, houve o cuidado de previamente a harmonizar e uniformizar, de maneira a que a mesma possa ser explorada convenientemente e de forma eficiente no Portal a ser construído.

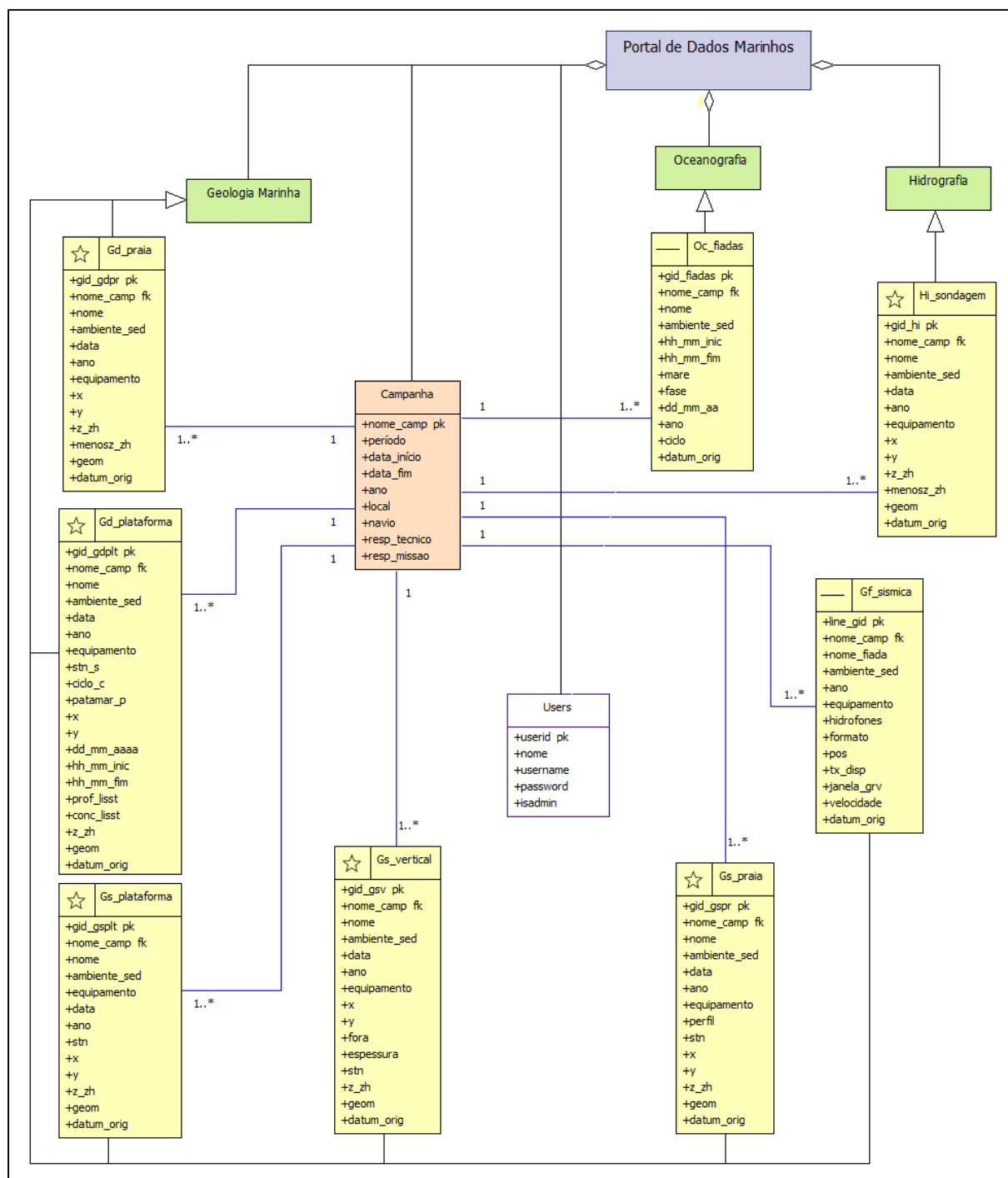





Figura 3.5 — Modelo de dados geográfico – diagrama de classes.

A **Tabela 3.3**, que se segue, foi editada com os campos considerados importantes para a compreensão e interpretação do diagrama de classes, em particular no que se refere ao nome atribuído às tabelas.

Tabela 3.3 — Elementos de apoio à interpretação do diagrama de classes.

	Cor	Prefixo	Ambiente Sedimentar	Área técnico-científica	
Entidades geográficas		Gd	Praia	Geologia Marinha	Amostras de sedimentos em suspensão
			Plataforma Continental		Amostras superficiais e amostras verticais de sedimento
		Gs	Praia		
			Gf		Plataforma Continental
		Hi	Praia	Hidrografia	
		Oc	Plataforma Continental	Oceanografia	
	Classes convencionais				
Superclasses					
Associação	 				
Relacionamentos					

O diagrama de classes foi criado com o modelo OMT-G, em linguagem UML, e apresenta as entidades geográficas “Gd_praia”, “Gd_plataforma”, “Gs_plataforma”, “Gs_praia”, “Gs_vertical”, “Gf_sismica”, “Hi_sondagem” e “Oc_fiadas” e as classes convencionais “Geologia Marinha”, “Hidrografia”, “Oceanografia”, “Campanha”, “Users” e “Portal de Dados Marinhos”. Todas elas entraram na criação da base de dados e no desenvolvimento do Portal de dados e definiram-se também os relacionamentos e a cardinalidade que existe entre estas entidades. A análise da **Figura 3.5** permite visualizar que a classe convencional “Portal de Dados Marinhos” é a agregação das entidades convencionais, “Geologia Marinha”, “Hidrografia”, “Oceanografia”, “Campanha” e “Users”. A classe convencional “Geologia Marinha” é uma generalização das entidades geográficas “Gd_praia”, “Gd_plataforma”, “Gs_plataforma”, “Gs_vertical”, “Gs_praia” e “Gf_sismica”, o que evidencia que esta classe é considerada a superclasse e as restantes as subclasses, sugerindo que todas têm características idênticas. Seguindo o mesmo raciocínio, pode-se referir que a classe convencional “Hidrografia” é a generalização da classe geográfica “Hi_sondagem”, o que significa que esta última entidade é uma especialização da classe convencional “Hidrografia”. Em relação à superclasse “Oceanografia”, tem-se a subclasse “Oc_fiadas”.

Neste diagrama, os tipos geométricos que predominam são o ponto e a linha, pelo que as subclasses “Gd_praia”, “Gd_plataforma”, “Gs_praia”, “Gs_plataforma”, “Gs_vertical” e “Hi_sondagem” estão definidas como ponto e a “Gf_sismica” e “Oc_fiadas” como linha. A leitura do diagrama mostra, ainda, que as ditas classes geográficas estabelecem ligação com a entidade convencional “Campanha”, o que permite transmitir a relação que, na realidade, existe entre estas entidades, levando a concluir que as associações estabelecidas ocorrem entre entidades convencionais e geográficas. Para cada uma delas, foi criada uma chave primária (*pk*) que tem como finalidade servir como identificador único de cada entidade e, para as classes geográficas que têm ligação com as convencionais, foram criadas chaves estrangeiras (*fk*), que fazem referência às chaves primárias destas últimas classes referidas. Em

relação às cardinalidades, pode-se constatar que as associações estabelecidas com a classe convencional “campanha” são que, numa campanha, existe pelo menos um elemento (1..*) de cada uma das entidades geográficas indicadas.

Este capítulo deu a conhecer, detalhadamente, os dados marinhos utilizados no desenrolar deste trabalho, tal como a estrutura do modelo de dados geográfico, apresentado através do diagrama de classes, que foi usado para implementar a base de dados espacial. Neste diagrama, estão indicados os metadados a serem explorados no portal pelo utilizador.

CAPÍTULO 4 – IMPLEMENTAÇÃO DO PORTAL

4.1. PostgreSQL/PostGIS

Em conformidade com as especificações do OGC para *Geographic Information – do Simple Feature Access – Part 2: SQL option*, a criação da base de dados espacial tem requisitos essenciais para as entidades geográficas, já que estas são compostas por uma coluna espacial que corresponde ao tipo de geometria da informação geoespacial (**Figura 4.1**). A coluna SRID (*Spatial Reference System Identifier*) é uma das mais importantes, uma vez que armazena o valor identificador de referência espacial, que determina a projeção e o *datum* a serem usados para exibir corretamente a geoinformação na superfície da Terra.

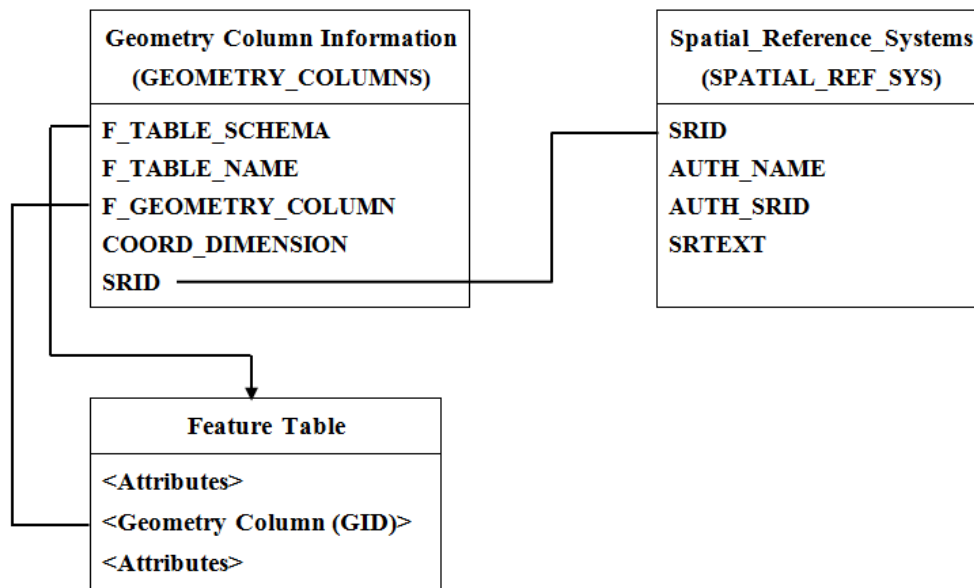


Figura 4.1 — Esquema para tabelas de *features*, usando SQL com tipos de geometria (adaptado de OGC, 2010).

De acordo com OGC (2010) e PostGIS (2017), indica-se o significado da terminologia usada na tabela:

➤ **GEOMETRY_COLUMNS:**

- *f_table_name* — utilizada para fazer referência ao nome da tabela que contém uma coluna de geometria espacial;
- *f_geometry_column* — usada para procurar a coluna da tabela onde estão as informações de coordenadas geográficas;
- *coord_dimension* — a dimensão espacial da coluna, em que podem ser considerados os valores 2, 3 ou 4;
- *srid* — *id* do sistema de referência espacial usado para a geometria de coordenadas na tabela específica, sendo, contudo, uma referência de chave estrangeira para a tabela *spatial_ref_sys*.

➤ **SPATIAL_REF_SYS:**

- *srid* — chave primária para a tabela de metadados “*spatial_ref_sys*” do OGC para as especificações do *Simple Features for SQL*;
- *auth_name* — nome do padrão citado para este sistema de referência, neste caso, o EPSG (*European Petroleum Survey Group*);

- **auth_srid** — *id* do sistema de referência espacial, conforme é definido pela autoridade citada no “auth_name”. Neste projeto, o *EPSG* tem o “id” 3763, que corresponde ao sistema de referência ETRS89/PT-TM06;
- **srtext** — representação *WKT* (*Well-Known Text*) do sistema de referência espacial;
- **proj4text** — coluna que contém a representação do *Proj4* (*Cartographic Projections Library*) do sistema de coordenadas para um determinado *SRID*. Convém referir que o *PostGIS* usa a biblioteca *Proj4* para fornecer as capacidades, a fim de efetuar a transformação de coordenadas de determinado *SRID* (PROJ.4, 2017).

Um identificador de sistema de referência espacial (*SRID*) é um valor único utilizado para reconhecer inequivocamente o código *EPSG* que, por sua vez, é definido como um conjunto de definições de sistemas de referência de coordenadas e transformações de coordenadas (*EPSG*, 2017). Estes códigos podem ser encontrados no endereço <http://www.epsg-registry.org/>. Os sistemas de referência espacial são definidos através da especificação *SFA - SQL* do *OGC*, usando o padrão *WKT*, pelo que os utilizadores podem consultar e visualizar os dados, gerar relatórios imprimíveis e criar padrões *WKT* compatíveis com a *ISO 19162* (*EPSG*, 2017). Neste âmbito, a facilidade no acesso a este repositório de sistemas proporcionou, aos fornecedores da solução *SIG* e às aplicações de *WebSIG*, proprietário e de código aberto, a possibilidade de adotarem o código *EPSG* para identificar o sistema de referência de informação geoespacial.

Quando se cria uma base de dados geográfica, com a extensão espacial *PostGIS*, são automaticamente construídas duas tabelas com os nomes “*spatial_ref_sys*” e “*geometry_columns*” (**Figura 4.2 a**) e **Figura 4.2 b**). De acordo com as especificações do *OGC* (2010), a primeira tabela é encarada como padrão para bases de dados espaciais e define os *SRID*, que são permitidos para as suas geometrias, e fornece informações sobre os correspondentes sistemas de referência. Por defeito, já vem com informação que permitirá realizar uma transformação de coordenadas em qualquer sistema de coordenadas que o utilizador solicitar.

Segundo *PostGIS* (2017), o *SRID* é um valor inteiro que identifica exclusivamente o sistema de referência espacial dentro da base de dados sendo, por outro lado, de acordo com *OGC* (2010), a chave primária para a tabela de metadados “*spatial_ref_sys*” do *OGC* para as especificações do *SFA - SQL*.

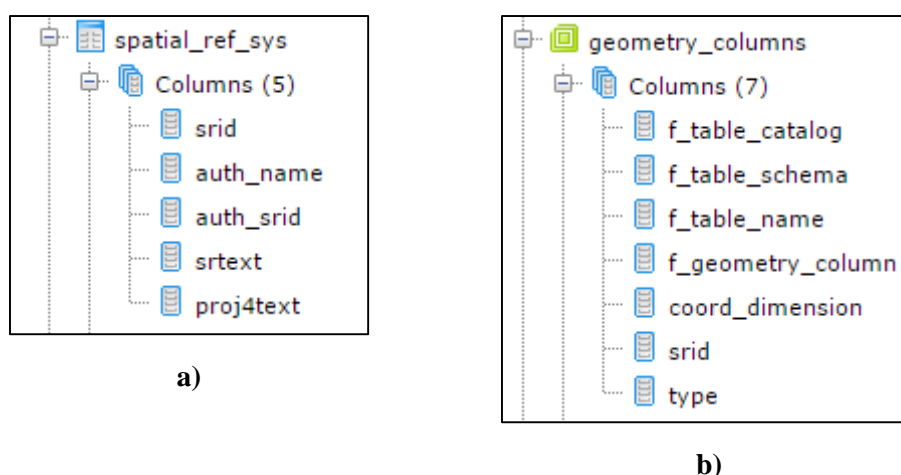


Figura 4.2 — Campos da tabela: a) *spatial_ref_sys*; b) *geometry_columns*.

Segundo as orientações da *OGC* para as especificações do *SFA - SQL*, os tipos de geometria *SQL* estão organizados conforme a estrutura hierárquica exposta na **Figura 4.3**, cujo diagrama de classes simplificado ilustra o modelo de dados específico para um elemento espacial.

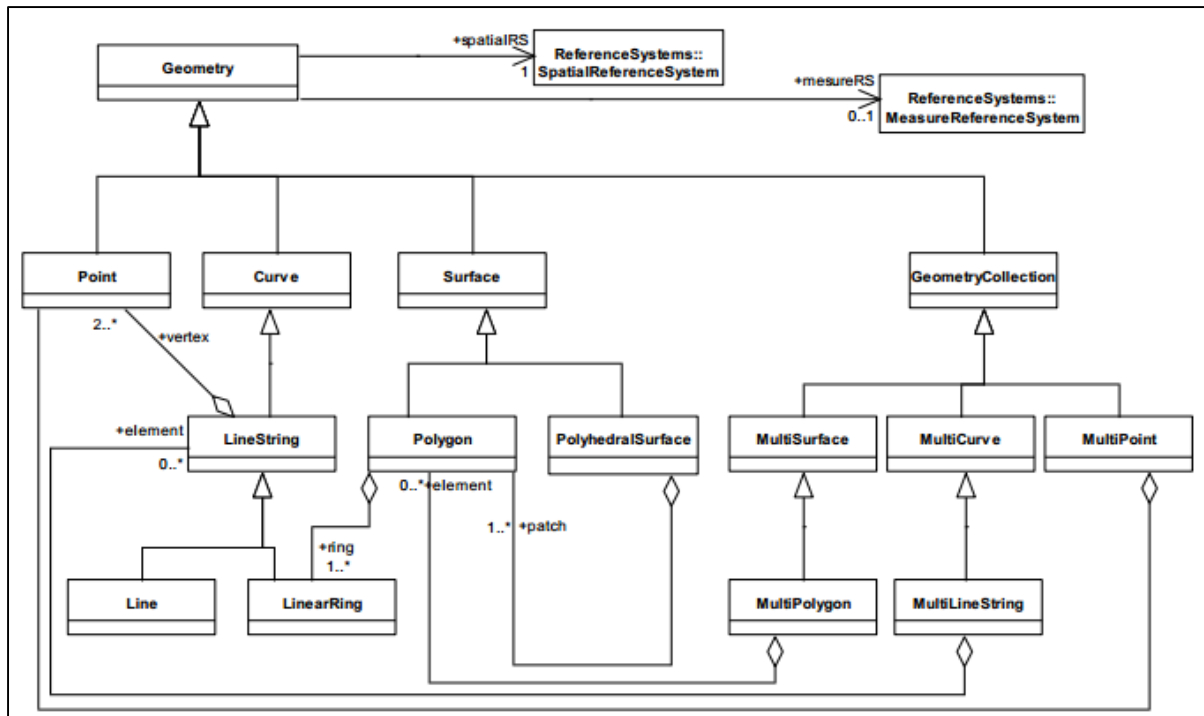


Figura 4.3 — Tipo de hierarquia geométrica em SQL (OGC, 2010).

É evidente que existe um elemento na “raiz” chamado **geometria**, que tem associado um sistema de referência espacial e que é uma generalização das entidades geográficas “*Point*”, “*Curve*”, “*Surface*” e “*GeometryCollection*”, o que indica que podem ser considerados subtipos da entidade “*Geometry*”. Por sua vez, a geometria tem como subclasses específicas as entidades “*MultiSurface*”, “*MultiCurve*” e “*MultiPoint*”, sendo de frisar que esta última é uma agregação da característica “*Point*”. Convém ainda alertar para o facto de serem necessários dois pontos para que a classe “*LineString*” seja uma agregação da classe “*Point*”.

Neste contexto, os tipos de restrições consideradas úteis para garantir a qualidade dos dados em bases de dados relacionais são:

- Restrições únicas.
- Não nulo.
- Restrições de chaves estrangeiras.
- Restrições de verificação de coluna.
- Restrições de verificação de tabela.

Entende-se assim que o *PostGIS* pode assegurar a qualidade dos dados espaciais, por via do uso comum e avançado da restrição *CHECK*, que tem como função validar os constrangimentos supra referidos.

Neste trabalho, pretende-se que o método de inserção dos dados, nas tabelas da base de dados, seja automático em vez de se introduzir a informação registo a registo. Esta alternativa é relevante por possibilitar que a alimentação da base de dados seja realizada de forma mais célere, otimizando esta etapa do projeto. Por outro lado, a introdução automática também reduz a possibilidade de engano na introdução dos registos. Na tentativa de alcançar este objetivo, recorreu-se à pesquisa de bibliografia e exploração de vários fóruns alusivos à temática em questão, tendo-se chegado à conclusão que a inclusão da informação espacial nas tabelas pode ser efetuada de duas maneiras distintas: 1) importação das *shapefiles* oriundas dos SIGs, neste caso, de programa proprietário (*ArcGIS*) ou 2) inserção direta dos registos. Para o efeito, existem várias maneiras conducentes à sua realização, tais

como:

- Comando usado na transformação de uma *shapefile* para SQL, passo que é realizado através do “*Command Prompt*”.

```
shp2pgsql.exe -s 3763 nome_ficheiro.shp nome_ficheiro > nome_ficheiro.sql
```

em que:

- **-s 3763** — indica o SRID, tendo sido utilizado o 3763;
 - **nome_ficheiro.shp** — é o nome do arquivo, que irá ser convertido para SQL;
 - **nome_ficheiro** — é o nome da tabela, que será criada na base de dados espacial;
 - **nome_ficheiro.sql** — é o nome do ficheiro SQL, que será executado para importar a *shapefile*.
- Ferramenta “Importar camada/ficheiro” — importa diretamente a *shapefile*, fazendo com que a tabela fique criada com a respetiva informação (**Figura 4.4**).

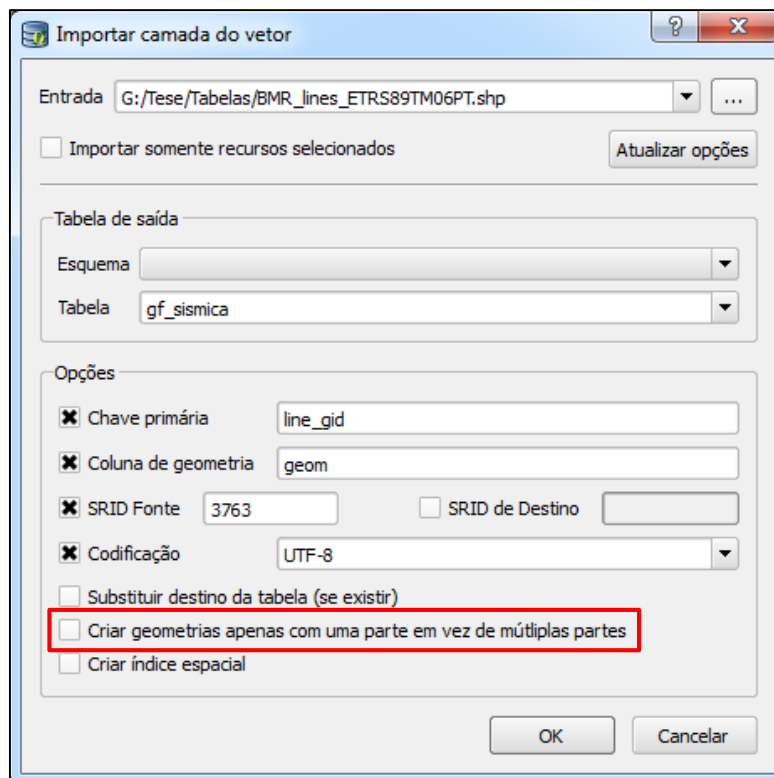


Figura 4.4 — Criação da tabela, através da importação de uma *shapefile*.

O retângulo de contorno vermelho mostra que esta tabela irá ser criada com o tipo de geometria *MULTILINESTRING*, uma vez que não se ativou esta opção.

- Função *AddGeometryColumn()*

O processo inicia-se com a criação da tabela, através do comando *CREATE TABLE*. Posteriormente, cria-se a coluna da geometria, utilizando o comando *SELECT AddGeometryColumn*, sendo a inserção dos valores alfanuméricos executada com o comando *INSERT INTO*. Este método pode ser explorado no *link* <https://postgis.net/docs/AddGeometryColumn.html>.

Uma das vantagens de inscrever os registos numa tabela ou coluna, usando a função

AddGeometryColumn() do *PostGIS*, é que esta adiciona automaticamente o “*CHECK*” nas restrições, à medida que vai testando a geometria inserida. Isto significa que verifica se o tipo de geometria está correto, se o SRID foi inserido e qual a dimensão da coluna espacial.

- Conjunto de comandos cuja finalidade é inserir a informação, através de ficheiros com extensão CSV de modo automático.

```
CREATE TABLE nome_tabela
Copy nome_da_tabela FROM 'caminho_do_ficheiro.csv' DELIMITER ',' CSV HEADER;
UPDATE nome_da_tabela
SET geom = ST_GeomFromText('MULTIPOINT(' || x || ' ' || y || ')',3763);
```

em que a:

- instrução *UPDATE* modifica os registos existentes numa tabela;
- função *ST_GeomFromText()* retorna uma instância de geometria da representação *WKT* do OGC.

```
ST_GeomFromText ( 'geometry_tagged_text' , SRID )
```

sendo:

- ***geometry_tagged_text*** a representação *WKT* da instância de geometria que se pretende retornar.

Face à explanação apresentada, optou-se por realizar o processo da criação da tabela e da inserção dos metadados para as entidades geográficas do tipo ponto, seguindo a última hipótese exposta. Sendo o carregamento mais rápido, em virtude da etapa da introdução da informação ser realizada automaticamente, as futuras atualizações serão facilitadas ao usar o mesmo processo. Esta realidade torna o trabalho muito mais célere e otimizado, em contraste com a inserção dos registos linha a linha, processo bastante moroso no caso de haver muitos registos. Evidencia-se o facto de, na opção escolhida, por analogia com o processo de importação direta de uma *shapefile*, não haver a limitação dos 10 caracteres no nome do campo, o que possibilita atribuir nomes mais sugestivos e perceptíveis. Para as entidades do tipo linha, o processo foi efetuado de acordo com a segunda hipótese, em que a tabela é criada diretamente, importando a *shapefile*.

Tal como já foi referido, ao criar uma tabela geográfica, é importante inserir constrangimentos referentes à chave primária, à dimensão, ao tipo de geometria, ao *id* do sistema de referência espacial e à chave estrangeira (**Figura 4.5**) para se aferir a qualidade dos dados espaciais. Contudo, para as tabelas convencionais, apenas se introduz a chave primária.

```
CONSTRAINT gd_praia_pkey PRIMARY KEY (gid_gdpr),
CONSTRAINT enforce_dims_geom CHECK (st_ndims(geom) = 2),
CONSTRAINT enforce_geotype_geom CHECK (geometrytype(geom) = 'MULTIPOINT'::text OR geom IS NULL),
CONSTRAINT enforce_srid_geom CHECK (st_srid(geom) = 3763),
FOREIGN KEY (nome_camp) references campanha(nome_camp)
```

Figura 4.5 — Constrangimentos inseridos aquando da criação da tabela.

em que:

- ***ST_NDims*** — retorna a dimensão da coordenada da geometria, pelo que o *PostGIS* suporta o valor 2 para o par coordenado (x,y).

- **Geometrytype** — indica o tipo de geometria de acordo com o já referido na especificação do OGC.

Na **Figura 4.6**, visualizam-se os constrangimentos inseridos aquando da criação da tabela, por forma a validar a qualidade dos dados espaciais, como já foi dito. O Anexo A apresenta a construção em SQL, das várias tabelas indicadas nas **Figura 4.7 a)** e **Figura 4.7 b)**.

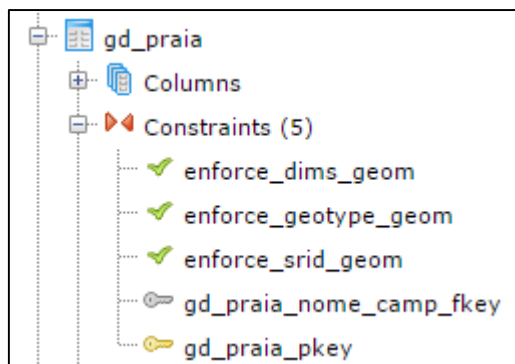





Figura 4.6 — Constrangimentos validados.

Os símbolos correspondentes aos constrangimentos têm o seguinte significado:

-  — Chave primária
-  — Chave estrangeira
-  — A geometria inserida está correta (CHECK)

Para as restantes entidades geográficas apresentadas no modelo de dados, seguiu-se o mesmo procedimento, tendo sido criadas as tabelas **Figura 4.7 a)** e **Figura 4.7 b)**:

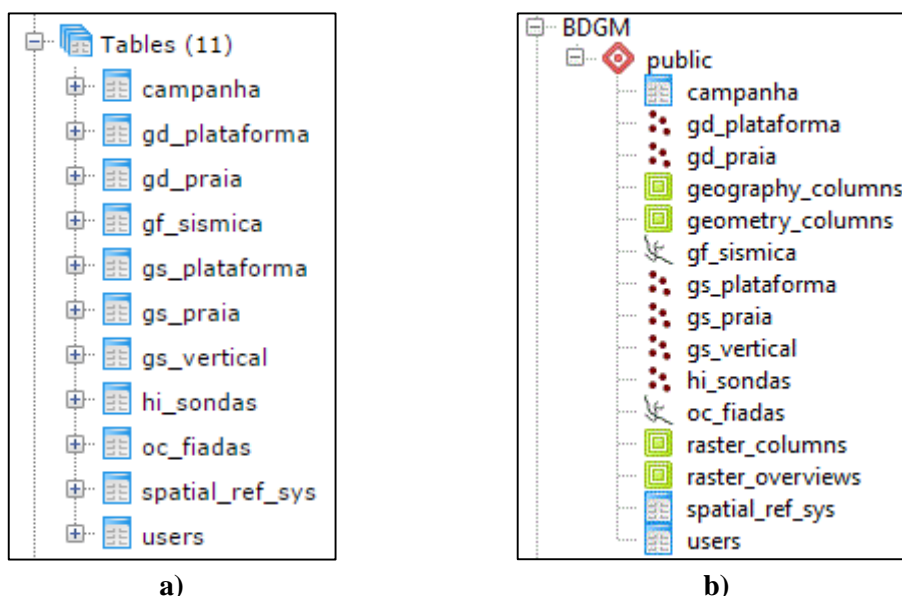
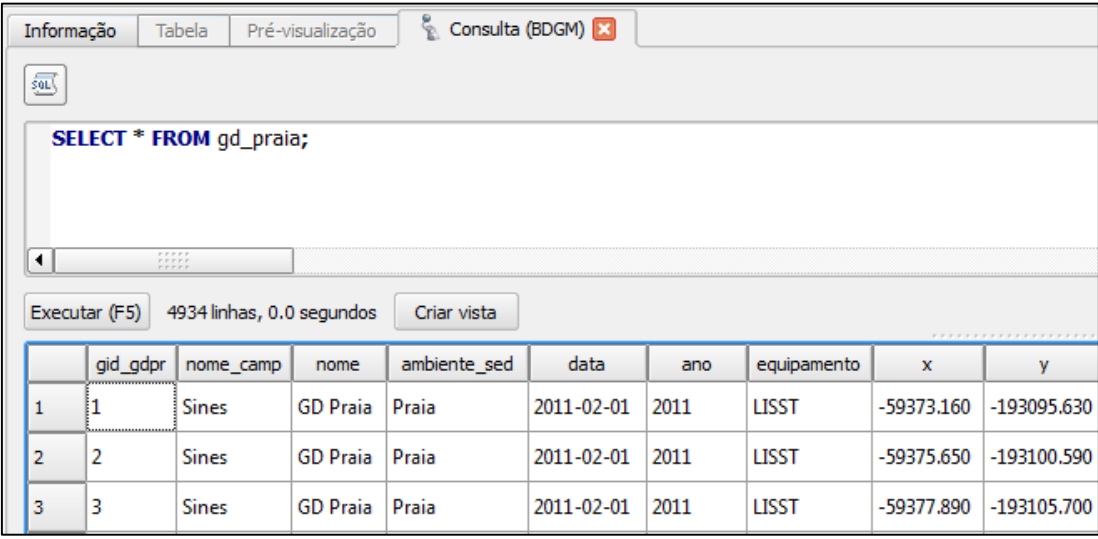


Figura 4.7 — Tabelas criadas no: a) *PostgreSQL*; b) *PostGIS*.

Terminado este procedimento, pode verificar-se se o número de registos foi inserido por completo e se o sistema de referência espacial é o correto (**Figura 4.8** e **Figura 4.9**). Estas duas situações são testadas por meio de duas *queries* em SQL com os seguintes códigos:


```
SELECT * FROM nome_da_tabela;
```



The screenshot shows a web-based SQL interface with tabs for 'Informação', 'Tabela', 'Pré-visualização', and 'Consulta (BDGM)'. The 'Consulta (BDGM)' tab is active, displaying a SQL query: `SELECT * FROM gd_praia;`. Below the query, it indicates 'Executar (F5) 4934 linhas, 0.0 segundos' and a 'Criar vista' button. The results are shown in a table with 10 columns: `gid_gdpr`, `nome_camp`, `nome`, `ambiente_sed`, `data`, `ano`, `equipamento`, `x`, and `y`. The first three rows of data are visible.

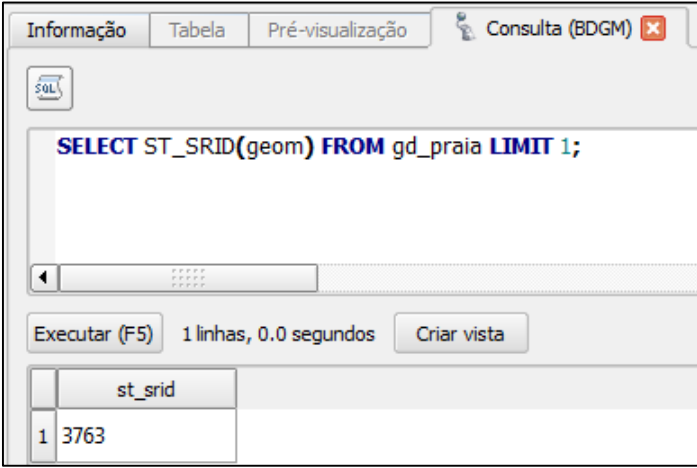
	gid_gdpr	nome_camp	nome	ambiente_sed	data	ano	equipamento	x	y
1	1	Sines	GD Praia	Praia	2011-02-01	2011	LISST	-59373.160	-193095.630
2	2	Sines	GD Praia	Praia	2011-02-01	2011	LISST	-59375.650	-193100.590
3	3	Sines	GD Praia	Praia	2011-02-01	2011	LISST	-59377.890	-193105.700

Figura 4.8 — Expressão de uma *query* em SQL e respetivo resultado (excerto).

```
SELECT ST_SRID(geom) FROM nome_da_tabela LIMIT 1;
```

em que:

- ❖ ***ST_SRID*** — devolve o identificador de referência espacial.
- ❖ ***LIMIT 1*** — retorna apenas o valor de um registo.



The screenshot shows the same SQL interface as Figure 4.8, but with a different query: `SELECT ST_SRID(geom) FROM gd_praia LIMIT 1;`. It indicates 'Executar (F5) 1 linhas, 0.0 segundos' and a 'Criar vista' button. The results are shown in a table with one column: `st_srid`. The first row of data is visible.

	st_srid
1	3763

Figura 4.9 — Função que devolve o identificador de referência espacial.

A função ***ST_Transform()*** restitui uma nova geometria com as coordenadas transformadas num sistema de referência diferente, sendo de sublinhar que a sua utilização só é possível com tipo de geometria “*Point*”, o que implica a conversão do “*MultiPoint*” para este. O processo pode ser realizado, através da função ***ST_DUMP()***, que tem como finalidade expandir multi-geometrias.

As funções ***ST_SRID()***, ***ST_DUMP()*** e ***ST_TRANSFORM()*** estão integradas nas especificações *Simple Feature Access SQL* do OGC (PostGIS, 2017).

4.2. Disponibilização da informação na internet

Esta etapa inicia-se com a instalação do pacote MS4W (*MapServer for Windows*), versão 3.2.1, composto pelo servidor *Web*, servidor de mapas e bibliotecas, procedimento que foi efetuado na plataforma *Windows 7*. O processo culmina com a criação de uma estrutura de pastas, que foram gravadas na diretoria *C:\ms4w*. Seguiu-se o *download* da *framework p.Mapper*, na versão 4.4.0 para ambiente *Windows*, visando instalá-la, tendo ficado alocada em *C:\ms4w\apps\pmapper*. Esta diretoria é válida para o desenvolvimento da aplicação deste caso concreto. De forma a confirmar o sucesso destes procedimentos, digita-se o endereço *http://localhost* no navegador, surgindo as imagens apresentadas nas **Figura 4.10 a)** e **Figura 4.10 b)**.

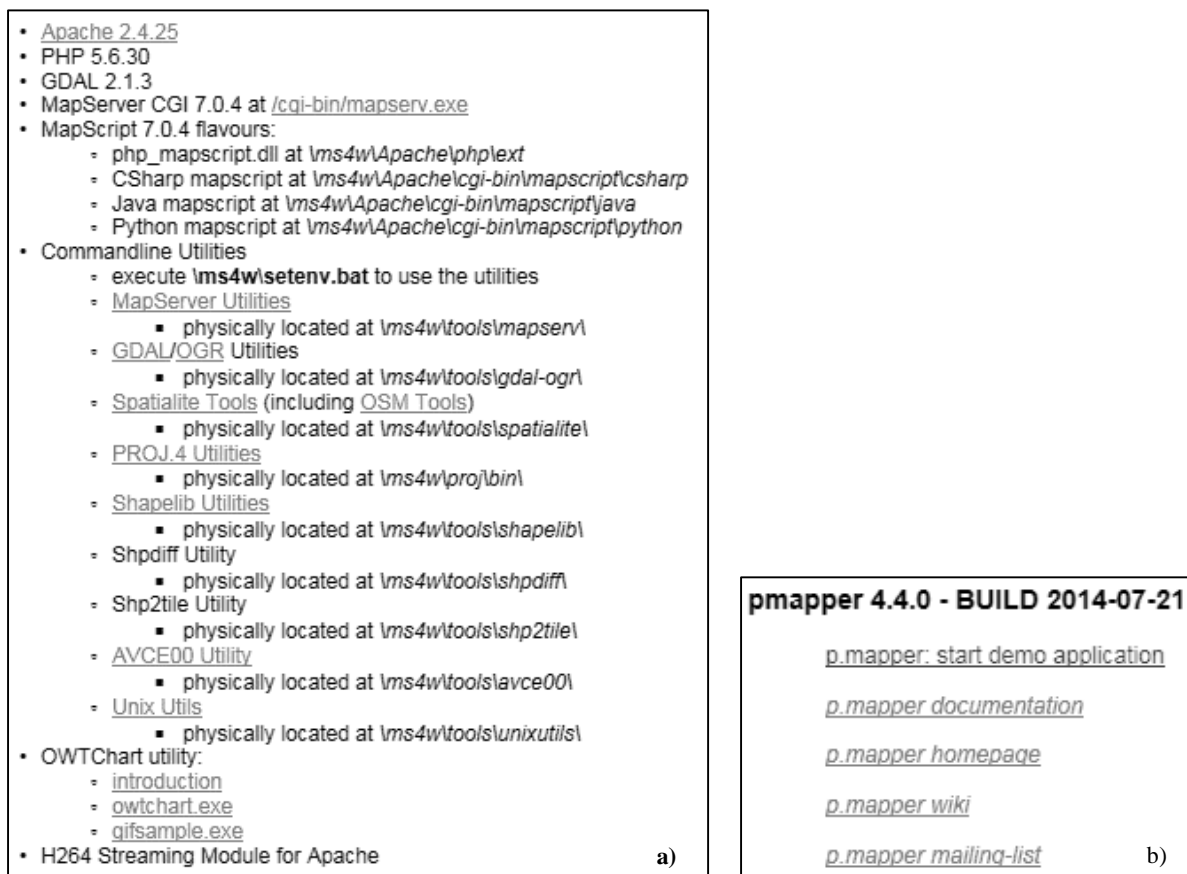


Figura 4.10 — Instalação: a) Pacote MS4W; b) *p.Mapper*.

4.2.1. MapServer

De acordo com Kanegae (2005), a estrutura de uma aplicação *MapServer* é composta por:

- **Dados** — elementos que devem estar num formato que seja lido pelo *MapServer*.
- **Mapfile** — ficheiro com extensão *.map*, em formato de texto ASCII (*American Standard Code for Information Interchange*) considerado o “coração” do *MapServer*, dado que é o ficheiro base para a configuração das aplicações. Tal facto baseia-se na estruturação do ficheiro, o que torna possível visualizar a informação geoespacial. É nele que se personaliza o tipo de informação, ao definir a maneira como os dados serão apresentados ao utilizador, através da configuração de determinadas características, tais como as cores, os símbolos, a legenda e a escala (**Figura 4.11**). O ficheiro *Mapfile* é um elemento importante na construção da aplicação, uma vez que está relacionado com a renderização da base de dados pelo servidor de mapas *MapServer* (Pimenta *et al.*, 2012). Esta circunstância torna-se relevante na medida

em que é neste ficheiro que são inseridos os vários objetos e parâmetros pré-definidos, por exemplo, a projeção cartográfica, o tipo geométrico da *layer* (*Point*, *Line*, *Polygon* e *Raster*) e outros parâmetros, que foram estabelecidos de forma a permitir a representação adequada da informação geográfica introduzida na base de dados. A simbologia é outro fator a valorizar na elaboração do *Mapfile*, dado que é por intermédio deste ficheiro que os objetos são apresentados na interface gráfica do utilizador, facilitando-lhe a visualização ao permitir distinguir as diversas camadas de informação. A **Figura 4.12** expõe os vários símbolos cartográficos disponíveis na aplicação *MapServer*.

- **Formulário de inicialização** — declaração em HTML, que tem como função enviar ao executável do *MapServer* (*mapserv.exe*) os parâmetros básicos para a inicialização da aplicação, designadamente, o caminho do *Mapfile* e o URL do *MapServer* CGI.
- **Template** — ficheiro que define a interface gráfica da aplicação, através do qual é definida a forma como o utilizador vai interagir com a aplicação e a maneira como os componentes gerados pelo *MapServer* (e.g. o mapa, a legenda, a escala) serão apresentados ao utilizador.

Neste âmbito, na **Figura 4.11**, é apresentada a estrutura de um ficheiro *Mapfile* adaptado para a presente dissertação.

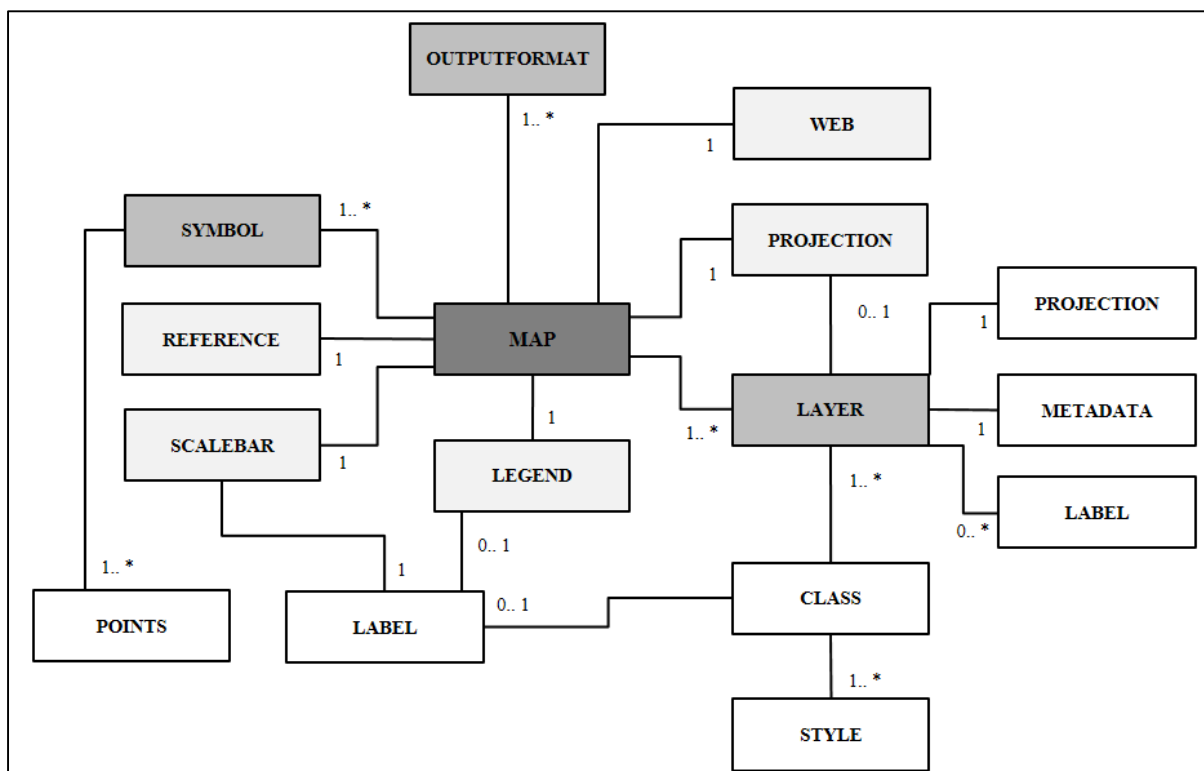


Figura 4.11 — Estrutura de um ficheiro *Mapfile*.

A sua análise permite sublinhar que a estrutura de um ficheiro *Mapfile* envolve vários elementos, que conduzirão à personalização da interface gráfica dos dados geográficos. A leitura deste diagrama leva a concluir que existem diversos tipos de relacionamentos entre os vários fatores, que são definidos com base na sua cardinalidade. Perante esta evidência, é de destacar os elementos que têm pelo menos um elemento (1..*) e que são o “*OUTPUTFORMAT*”, a “*LAYER*” e o “*SYMBOL*”, uma vez que são os que irão provocar mais dinamismo na aplicação *Web*. Todavia, além destes, existem outros que estão ligados diretamente ao elemento principal “*MAP*” tais como, “*WEB*”, “*PROJECTION*”, “*LEGEND*”, “*SCALEBAR*” e “*REFERENCE*”, casos em que existe uma única ocorrência deste fator.

As principais variáveis que compõem o ficheiro *Mapfile* são:

- **MAP** — que define o início do *Mapfile*.
- **WEB** — composto pelo caminho absoluto do *template* e do ficheiro de inicialização.
- **PROJECTION** — que determina o sistema de referência em que será visualizado o mapa. A configuração das projeções pressupõe que sejam definidas duas, uma para a imagem de saída (no objeto MAP) e outra para cada camada (na *layer*) a ser projetada.
- **LEGEND** — em que é descrita a forma como a legenda é construída.
- **LAYER** — em que são delineadas as camadas de informação geográfica a representar. O número de camadas corresponderá ao número de entidades, que estão apresentadas no modelo de dados já descrito, sendo nesta variável que se inicia um objeto *layer*.
- **NAME** — que define o nome da camada de informação para a aplicação, que não pode ser constituído por espaços nem acentos e deve ser uma *string* com um máximo de 20 caracteres. Este nome terá que coincidir com o que é inserido no ficheiro *config_default.xml* na *tag* das categorias.
- **CLASS** — que indica o início da declaração deste elemento, em que estão inseridas as variáveis “STYLE” e “OUTLINECOLOR” e termina com um “END”. Todas as *layers* com representação vetorial devem ter pelo menos uma “*class*”.
- **STYLE** — que contém simbologia e *design*, podendo aplicar-se diversos estilos dentro de uma “*CLASS*” ou “*LABEL*”.
- **QUERYMAP** — que permite personalizar o resultado obtido pela pesquisa, através de uma *query*, por exemplo, este pode ser apresentado de uma forma destacada, com uma cor diferente ou os elementos desenhados de acordo com o que foi estipulado para a camada de informação.
- **LABEL** — usada para anotar um recurso com um pedaço de texto.
- **SYMBOL** — variável através da qual são definidos os símbolos, que podem ser usados na construção de uma *layer*, para descrever os elementos que são do tipo ponto (**Figura 4.12**).

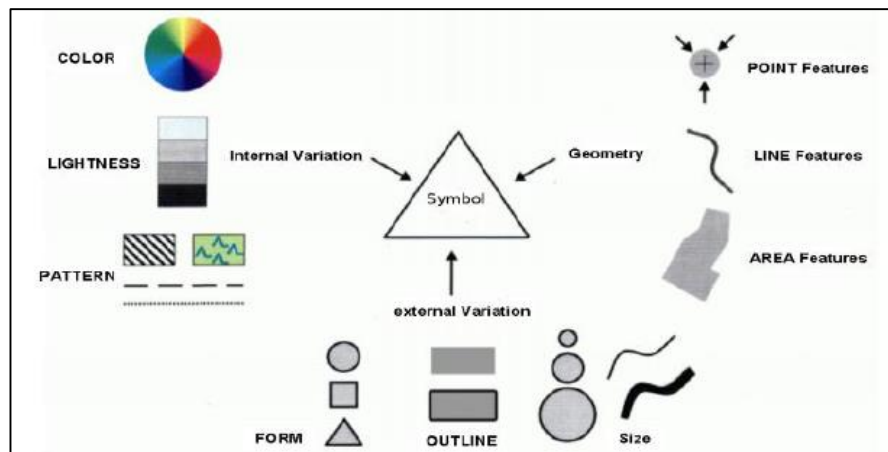


Figura 4.12 — Estrutura de um *Mapfile* (Freimuth *et al.*, 2017).

- **POINTS** — que indica o início de uma sequência de pontos, que compõem um símbolo do tipo vetor.
- **REFERENCE** — que permite mostrar a extensão de um mapa numa interface interativa.
- **OUTPUTFORMAT** — que define os formatos de saída disponíveis que são suportados, como o PNG (*Portable Network Graphics*), o GIF (*Graphics Interchange Format*), o JPEG (*Joint Photographic Experts Group*), entre outros.
- **SCALEBAR** — que permite a criação da escala gráfica.

Existem outros elementos de menor relevância, que também importa mencionar, para se ficar com um bom entendimento de um ficheiro *Mapfile*, são eles:

- **TYPE** — define o tipo geométrico da *layer* a representar, que pode ser do tipo ponto, linha e polígono, no caso de informação vetorial e do tipo *raster*, se a informação for matricial.
- **SIZE** — define as dimensões do elemento no mapa interativo.
- **OUTLINECOLOR** — define a cor da camada de informação, por meio de um código RGB (Red, Green, Blue).
- **COLOR** — indica a cor da *layer*.
- **EXTENSÃO** — define a área envolvente do mapa na inicialização da aplicação.

Feito o esclarecimento da função das variáveis (**Figura 4.11**) que compõem o ficheiro *Mapfile* e dando especial atenção aos elementos que derivam da variável *layer*, apresenta-se, na **Figura 4.13**, o exemplo das linhas de código inseridas para a criação de uma camada de informação no referido ficheiro.

```
#
# Start of Layer Águas Praia
#
LAYER
  NAME "gd_praia"
  TYPE point
  CONNECTIONTYPE POSTGIS
  PROCESSING "CLOSE_CONNECTION=DEFER"
  CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
  DATA "geom FROM gd_praia USING UNIQUE gid_gdpr USING SRID=3763"
  TOLERANCE 6
  TOLERANCEUNITS pixels
  TEMPLATE void
  PROJECTION
  : "init=epsg:3763"
END
METADATA
  "DESCRIPTION" "Águas Praia"
  "RESULT_FIELDS" "nome_camp, x, y, ano, equipamento, z_zh, menosz_zh, datum_orig"
  "RESULT_HEADERS" "Campanha, X, Y, Ano, Equipamento, Z_ZH, MenosZ_ZH, Datum_origem"
  "LAYER_ENCODING" "UTF-8"
END # Metadata
CLASS
  #Name 'Águas Praia'
  SYMBOL 'circle'
  SIZE 5
  COLOR 180 180 180
END # Class
END # Layer
```

Figura 4.13 — Exemplo da criação de uma *layer* no *Mapfile*.

O retângulo com o contorno a laranja indica as linhas de código necessárias para estabelecer a ligação à base de dados espacial *PostgreSQL/PostGIS*, tal como para extrair a informação geoespacial e alfanumérica referente à tabela especificada. A linha **PROCESSING "CLOSE_CONNECTION = DEFER"** mostra que, a primeira vez que for efetuada a conexão à base de dados, esta será partilhada por todas as camadas de informação que usam os mesmos elementos de ligação, no que respeita ao *host*, à base de dados, ao *user* e à *password*. Tal evidência possibilita uma melhor renderização à base de dados, já que a ligação estabelecida com esta não é fechada, o que permite uma maior velocidade.

No que respeita ao retângulo limitado a azul, as linhas de código têm o seguinte significado:

- **DESCRIPTION** — nome da *layer* que aparece na legenda da aplicação.
- **RESULT_FIELDS** — nome dos campos na base de dados.
- **RESULT_HEADERS** — nome dos campos a visualizar na tabela, com o resultado de uma *query* e que são referentes aos nomes indicados em **RESULT_FIELDS**.
- **LAYER_ENCODING** — o código UTF-8 é um tipo de codificação binária (*Unicode*), que pode ser de 1 a 4 *bytes* de comprimento e pode representar qualquer carácter no padrão *Unicode*. É um código *multibyte*, isto é, usa vários *bytes* por carácter e é compatível com o *ASCII*.

É importante que o elemento “**TEMPLATE void**” seja inserido, porque é um fator que indica que a camada de informação pode ser interrogável, o que significa que pode ser consultada, através de uma *query* em SQL.

4.2.2. p.Mapper

Esta aplicação tem ficheiros que são considerados essenciais no sentido de requerem alteração, de modo a personalizar a plataforma com a informação pretendida, que se descrevem a seguir:

pmapper_demo.map

O *Mapfile* (*.map) é o ficheiro no qual estão inseridas as configurações a visualizar no mapa interativo, tendo em vista satisfazer os interesses de cada entidade para gerar o mapa que irá ser visto. Sublinha-se que este ficheiro deverá ser o primeiro a alterar, em conformidade com os parâmetros já referidos e necessários à personalização da aplicação (Anexo B).

config_default.xml

Este ficheiro é constituído por diversas variáveis e parâmetros, que foram alterados de forma a atingir o expectável, indo ao encontro dos objetivos deste estudo. É nele que estão inseridas as configurações de todas as *tag*, que são as estruturas de linguagem de marcação que contém instruções (exemplo na **Figura 4.14**), e que serão utilizadas para personalizar o *WebSIG*:

- **plugin** (export, scalebar, etc);
- **config** — nesta *tag*, são indicadas as pastas onde estão guardados os ficheiros *Mapfile* e os *scripts* em *javascript*, assim como o caminho do ficheiro com extensão *xml*, onde estão definidos os parâmetros de impressão;
- **map** — em que são indicados o nome dos ficheiros *Mapfile* e *Template* sendo que, relativamente a este último, ainda é referido o local onde está alojado;
- **categories** — as *layers* podem ser subdivididas em diferentes categorias (neste caso, Geologia Marinha, Hidrografia e Oceanografia) e, em cada categoria, são inseridas as *features*, que vão ser apresentadas na TOC (*Table Of Contents*), para visualização (**Figura 4.14**);
- **query** — *tag* em que são especificados todos os parâmetros relacionados com a consulta da informação espacial, por exemplo, o número de registos que serão extraídos e a cor dos elementos que forem selecionados;
- **user interface (ui)** — é a configuração dos parâmetros afetos à legenda, como é o caso do estilo das categorias (em árvore ou planas), a necessidade de uma “checkbox” para as *layers* e para as *features*, tal como a opção da lista de camadas (TOC) ser atualizada automaticamente, de acordo com a escala;
- **locale** — onde é indicada a língua em que será apresentada a interface gráfica, bem como o tipo de caracteres (UTF-8);
- **print** — em que são definidos os formatos de saída das imagens;
- **download** — *tag* em que é possível indicar os níveis de DPI (*Dots per inch*) para gravação das imagens.

```

<categories>
  <category name="Amostragem">
    <group>gd_praia</group>
    <group>gd_plataforma</group>
    <group>gs_plataforma</group>
    <group>gs_vertical</group>
    <group>gs_praia</group>
  </category>
  <category name="Levantamento Geofísico">
    <group>sismica</group>
  </category>
  <category name="Levantamento Hidrográfico">
    <group>sondas</group>
  </category>
  <category name="Levantamento Oceanográfico">
    <group>oc_fiadas</group>
  </category>
  <category name="Raster">
    <group>orto2010</group>
    <group>mdt</group>
    <group>sed6</group>
  </category>
  <category name="Outros">
    <group>portugal</group>
  </category>
</categories>

```

Figura 4.14 — Categorias.

js_config.php

É um ficheiro que se encontra fisicamente em *C:\ms4w\apps\pmapper\pmapper-4.4.0\config\default*, sendo constituído por linhas de código em linguagem de programação PHP, com a função de definir a configuração de ferramentas, entre outras: os botões a representar na barra de ferramentas, a configuração da escala gráfica, a opção de escolha do resultado da *query* ser ou não apresentado numa tabela. No caso em estudo, este ficheiro foi sendo editado com a remoção de algumas ferramentas e a inclusão de outras, à medida que os *plugins* iam sendo adicionados no ficheiro *C:\ms4w\apps\pmapper\pmapper-4.4.0\config\config_default.xml*.

Além destes ficheiros principais, existem ainda outros que também contribuem para a personalização da aplicação *Web*, especificamente: os *templates* CSS, que permitem a estilização da página (*C:\ms4w\apps\pmapper\pmapper-4.4.0\templates*) e os *templates* em PHP, que podem ser alterados de um modo subjetivo, consoante a necessidade do utilizador e o seu grau de conhecimento e domínio da linguagem de programação (*C:\ms4w\apps\pmapper\pmapper-4.1.1\incphp*). Neste projeto, não houve necessidade de modificar estes ficheiros para alcançar o objetivo final.

A aplicação *p.Mapper* tem como princípio o *template* de inicialização *index.phtml*, que está armazenado em *C:\ms4w\apps\pmapper\pmapper-4.4.0*, cujo conteúdo é a indicação, em linguagem de programação PHP, do ficheiro que dá acesso à interface gráfica, *map_default.phtml*, estando alocado na mesma diretoria (Figura 4.15). O ficheiro *uiement.php*, alojado em *C:\ms4w\apps\pmapper\pmapper-4.4.0\incphp*, foi modificado com a inclusão do logótipo do Instituto Hidrográfico e com o título “Portal de Dados Marinhos”, no cabeçalho.

A interface gráfica do Portal em curso está estruturada em cinco áreas distintas: o cabeçalho, o mapa interativo (área de visualização), as camadas de informação (*layers*), o mapa de referência e as ferramentas (Figura 4.15).

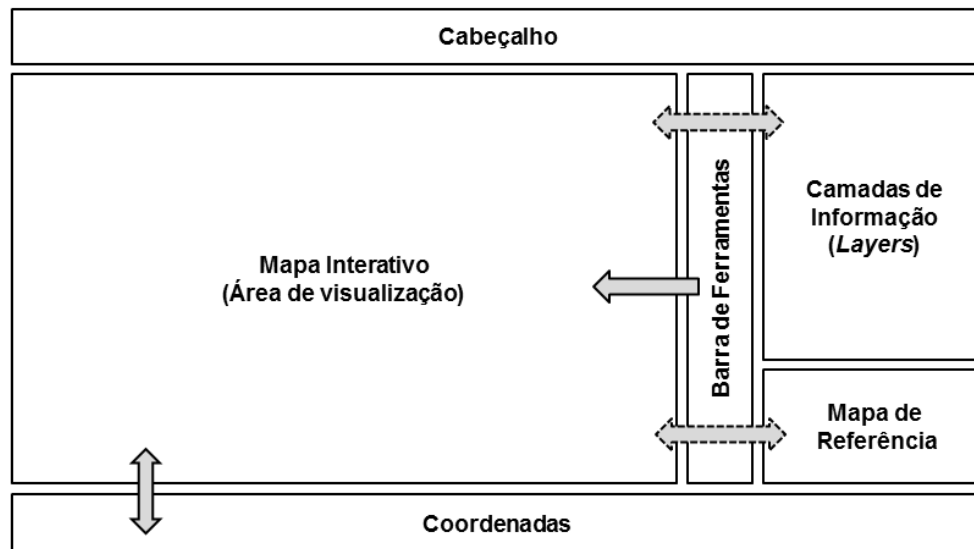


Figura 4.15 — Organização da Interface Gráfica.

Em sintonia com o modelo de dados geográfico apresentado, criou-se uma classe convencional “users”, em que foram inseridos os utilizadores com permissão para aceder ao “Portal de Dados Marinhos”, com os respetivos *username* e *password*. A Figura 4.16 esquematiza o raciocínio lógico, que sustenta o processo de autenticação criado neste caso. Convém referir que a classe convencional “users” se traduz na tabela com o mesmo nome, construída na BDG, e que o *script* criado, para este processo, foi efetuado em linguagem de programação PHP. Neste código, apresentado no anexo C, à semelhança do que já foi referido para o *Mapfile*, foi efetuada a conexão ao *PostGIS* com a indicação dos parâmetros de acesso. Em seguida, foi criada uma *query* em SQL (*\$sql = "SELECT * FROM users WHERE username = '\$username' AND password = '\$password'";*), que valida as credenciais introduzidas pelo utilizador (Figura 4.16). O ficheiro *index.phtml* foi alterado de forma a que a aplicação inicie com o sistema de autenticação (Figura 4.17), quando o Portal é carregado, em vez de ir diretamente para a interface gráfica do utilizador, como acontecia por defeito (ficheiro *map_default.phtml*).

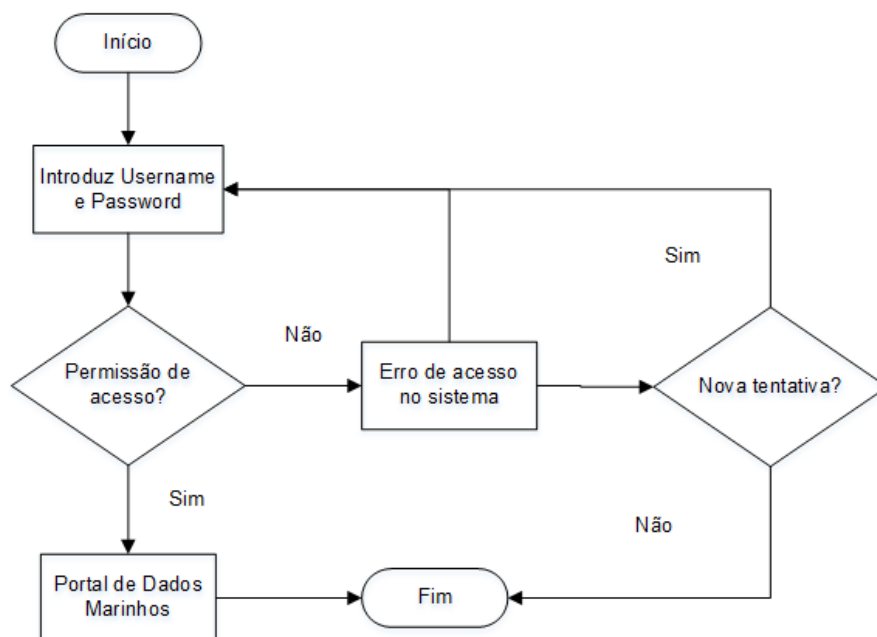
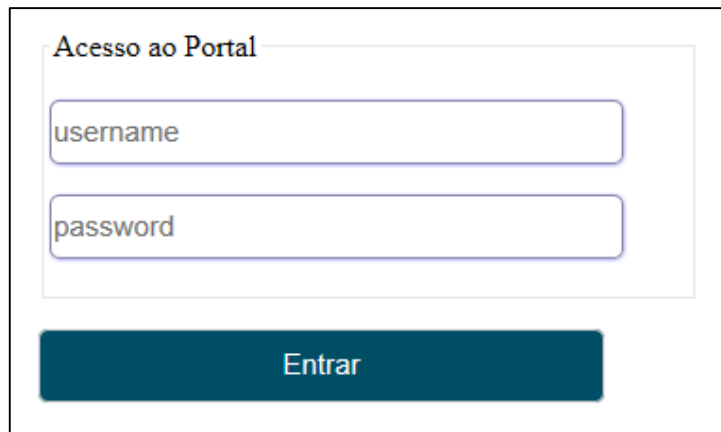


Figura 4.16 — Esquema de autenticação.

A análise da figura permite concluir que o utilizador, que tiver o *username* e *password* validado pela BDG, tem acesso ao Portal de Dados Marinhos (**Figura 4.17**). Este procedimento é verificado, depois de realizada a *query*, com o comando *if-else* em que: se a condição for verdadeira, o ficheiro *map_default.phtml* é lido e o utilizador tem acesso direto à interface gráfica; caso contrário, houve um erro de acesso ao sistema (*Your username or password is incorrect!*), o que implica tentar novamente, ou o acesso é negado por falta de autorização.



Acesso ao Portal

username

password

Entrar

Figura 4.17 — Sistema de autenticação.

A aplicação foi construída e estruturada com a finalidade de proporcionar ao utilizador navegar e efetuar pesquisas, usufruindo de ferramentas apropriadas para dar resposta ao que for solicitado. Deste modo, o diagrama apresentado na **Figura 4.18** explica, sucintamente, os papéis a desempenhar pelo utilizador e pelo administrador criados nesta aplicação.

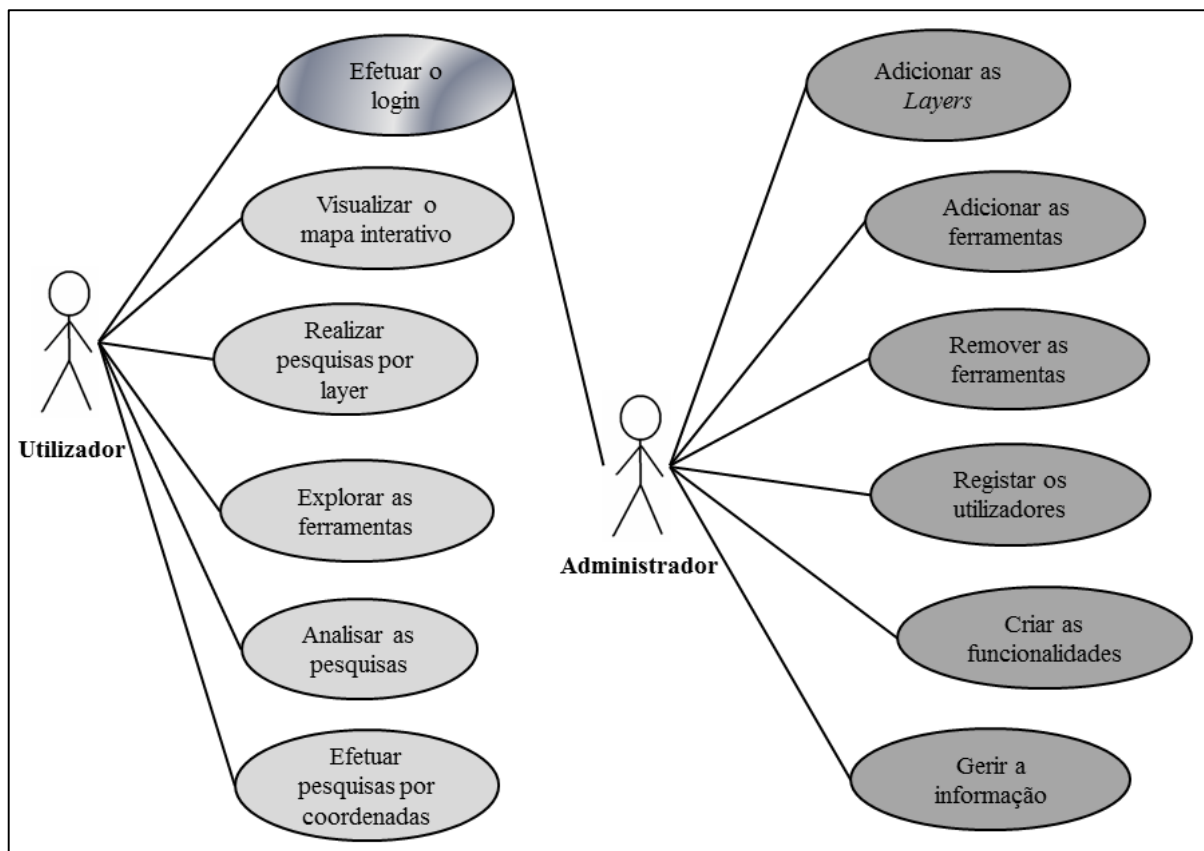


Figura 4.18 — Caso de uso para este protótipo.

Verifica-se que a função do utilizador é explorar a informação, enquanto que o papel do administrador é proporcionar as condições necessárias para que tal se concretize. Assim, este último tem como tarefas adicionar as *layers*, inserir e remover as ferramentas, registar os utilizadores, criar as funcionalidades e gerir a informação. Quanto ao utilizador, é-lhe dada a oportunidade de efetuar o login, visualizar o mapa interativo, realizar e analisar as pesquisas por *layer*, explorar as ferramentas e efetuar pesquisas, inserindo as coordenadas da área envolvente.

A **Figura 4.19** mostra a interface gráfica, que surge no seguimento da **Figura 4.17** em que, como padrão, apenas estão representadas as *layers* referentes a Portugal Continental, linha de costa e território emerso na zona periférica do porto de Sines, e ao modelo batimétrico deste porto. Posteriormente, o utilizador tem a liberdade de introduzir as camadas de informação que pretender para atingir o seu objetivo.

Com esta aplicação, o utilizador tanto pode visualizar uma camada de informação independente das outras como várias camadas integradas o que, neste último caso, lhe permite observar a sobreposição espacial das várias *layers*, oriundas de áreas distintas com diferentes tipos de formato. Perante isto, há a garantia de que esta aplicação permite a sobreposição de informação geográfica em formatos vetorial e matricial, constituindo, deste modo, uma apresentação híbrida. Aproveitando esta potencialidade, previu-se a possibilidade de se selecionar, através de um *click*, as camadas de informação que se pretende visualizar, dentro da respetiva *checkbox*, de modo a ativar ou a desativar a *layer* (**Figura 4.20**).

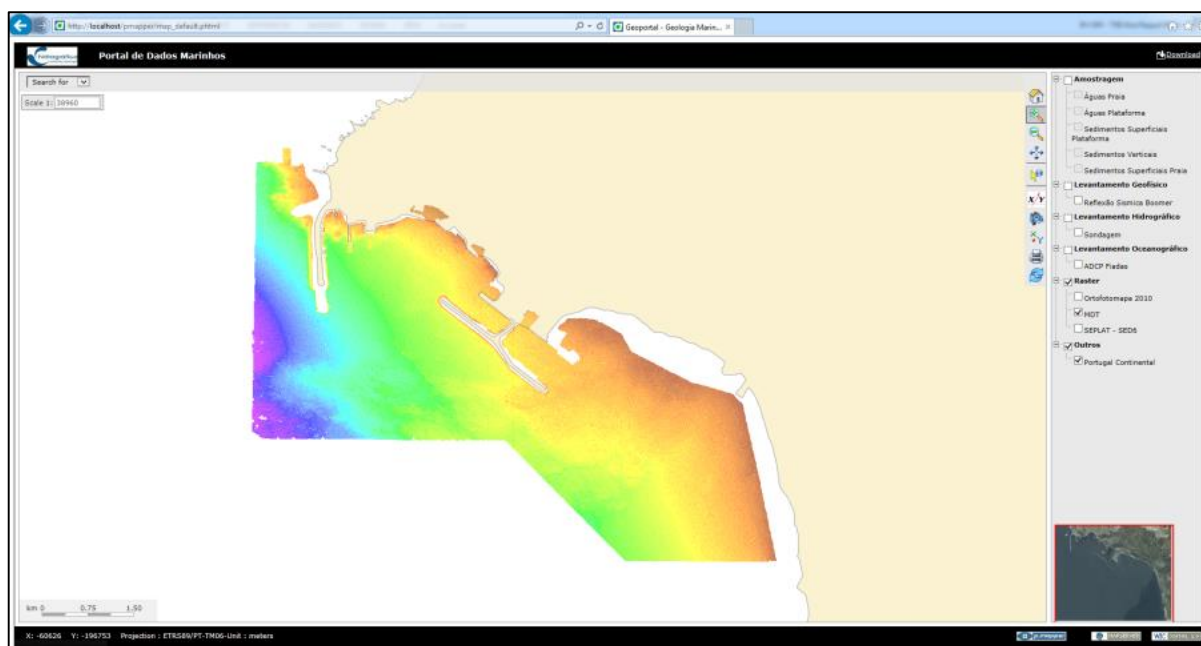


Figura 4.19 — Interface gráfica com as *layers* iniciais, Linha de costa, Território emerso e o Modelo batimétrico.

A informação espacial, apresentada no modelo de dados geográfico, pode ser visualizada na **Figura 4.21**, através da ativação de todas as *layers*, por intermédio do mapa interativo.

A função do mapa de referência é ir mostrando ao utilizador a sua posição relativa, à medida que vai navegando no mapa, sendo a área visível deste marcada por uma borda vermelha (**Figura 4.22 a**) e **b**). Realça-se a vantagem deste mapa estar sincronizado com a área de visualização (mapa interativo), proporcionando um enquadramento espacial da informação que está a ser apresentada (**Figura 4.22 c**) e **Figura 4.22 d**). Este mapa foi configurado no ficheiro *Mapfile* (*C:\ms4w\apps\pmapper\pmapper-4.4.0\config\default*), apresentado no anexo B, e a imagem, em formato PNG, foi guardada em *C:\ms4w\apps\pmapper\pmapper-4.4.0\images*.

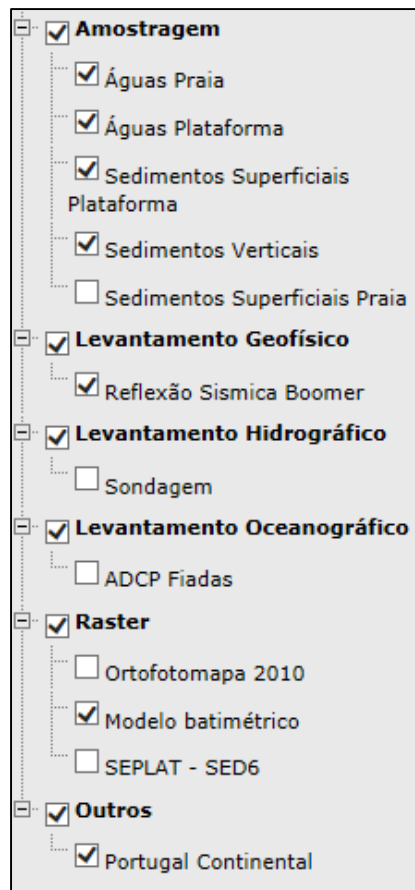


Figura 4.20 — Lista das camadas de informação.

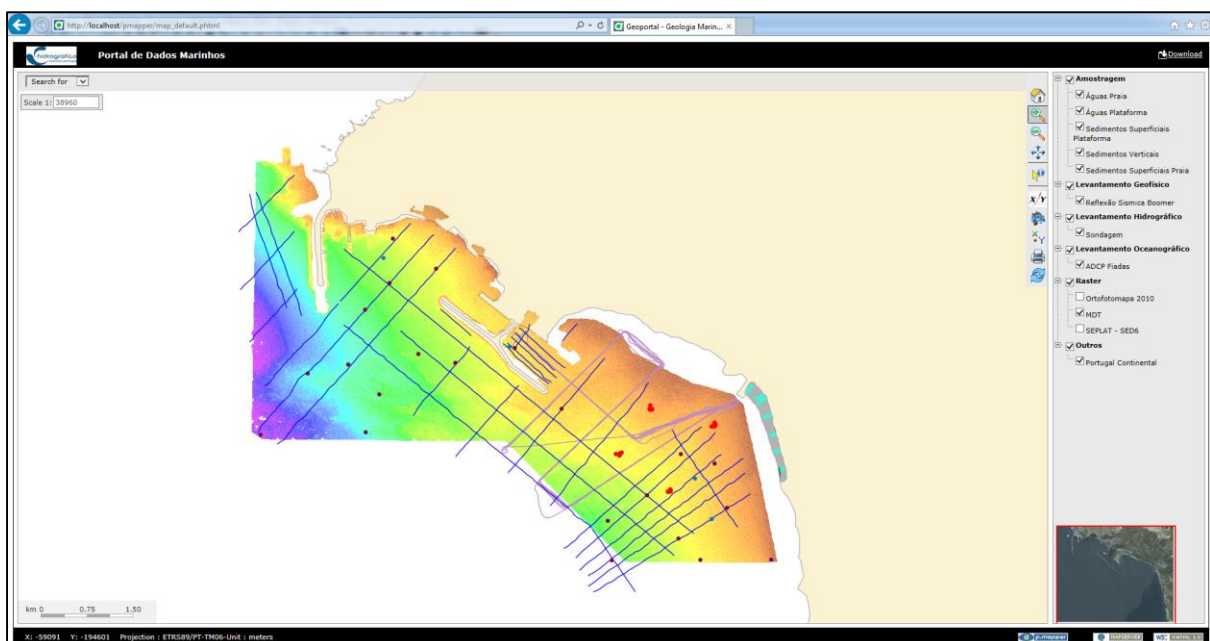


Figura 4.21 — Interface gráfica com todas as camadas de informação ativas.

O servidor de mapas foi personalizado em função dos interesses definidos para este projeto, tendo em vista que a plataforma permita que os utilizadores interajam de uma forma intuitiva, simples e acessível aos metadados pretendidos. Na **Tabela 4.1**, são apresentados e descritos os vários tipos de botões e respetivas funcionalidades.

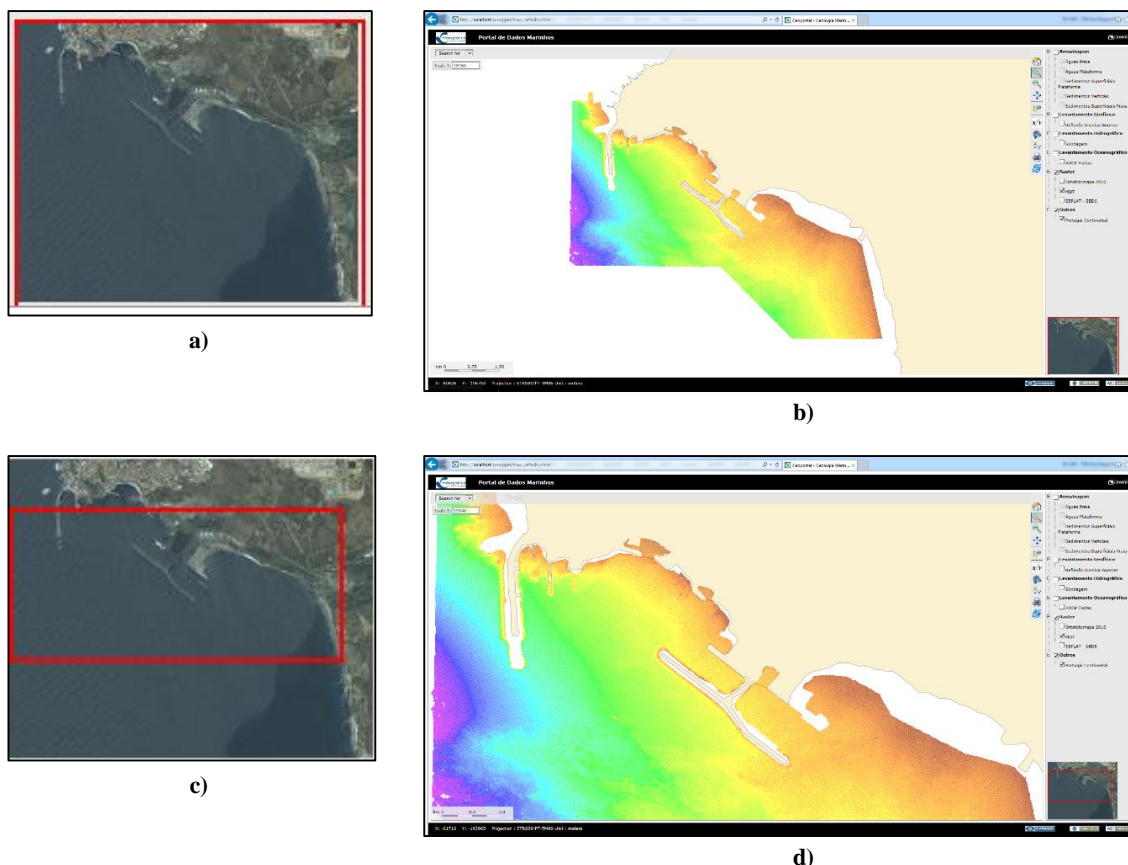













Figura 4.22 — Mapa de referência e área de visualização: a) e b) extensão inicial do mapa; c) e d) escala de representação maior.

Tabela 4.1 — Composição da barra de ferramentas.

Barra de Ferramentas			
Navegação	Zoom		Zoom in — para ampliar uma área específica do mapa, através de um <i>click</i> do rato ou de um retângulo da área que se pretende ampliar.
			Zoom out — permite um afastamento do mapa, com um <i>click</i> do rato, sobre a área que se pretende reduzir.
	Atualização		Zoomfullextent — tem a função de voltar para a extensão inicial do mapa, de modo a visualizar-se toda a informação geográfica inserida na aplicação.
			Reload — deixa apagar a seleção de uma <i>feature</i> no mapa interativo.
	Translação		Pan — move o mapa para o local pretendido, através de um <i>click</i> do rato sobre o mesmo.
	Interrogação		Identify — viabiliza a extração da informação alfanumérica associada a um elemento, clicando em cima do mesmo.
	Pesquisa		Query Editor — permite efetuar uma pesquisa sobre os dados geoespaciais que se pretende.
			Query — oferece a possibilidade de efetuar uma <i>query</i> apenas para a área envolvente que se deseja, através da inserção de coordenadas.
	Localização		Locate XY — deixa navegar até ao ponto coordenado.
	Impressão		Permite obter o mapa impresso nos formatos analógico, PDF ou TIFF, com a respetiva legenda.

Conforme já referido, o botão de interrogação *Identify* () permite ao utilizador seleccionar o ponto de interesse e consultar a informação que lhe está associada. Para tal, deve ativar-se o botão e clicar-se em cima da *feature*, o que faz com que a aplicação abra uma nova janela com uma tabela em que está a informação alfanumérica associada à *layer* escolhida. Salienta-se que é possível exportar esta tabela para o formato PDF (*Portable Document Format*), CSV (*Comma-separated values*) ou *Shapefile* (**Figura 4.23**).

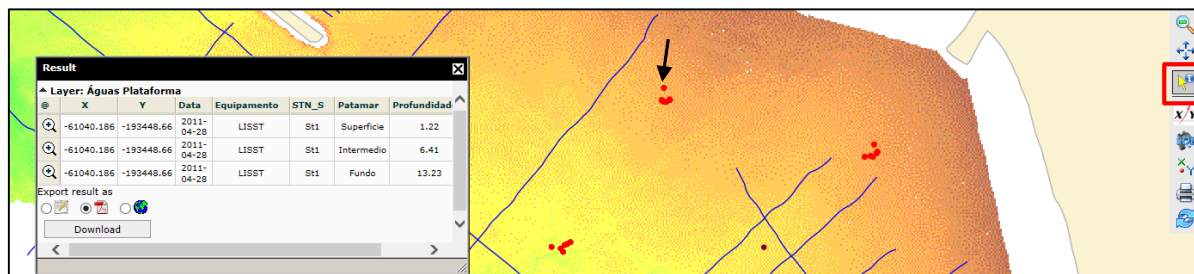



Figura 4.23 — Resultado da ferramenta *identify*.

A ferramenta a que corresponde o botão () foi construída com a pretensão de permitir ao utilizador obter apenas os metadados que lhe interessa, através da inserção das coordenadas da área envolvente que pretende. Neste sentido e com o propósito de a pesquisa ser mais específica, foram adicionados alguns atributos, que restringem ainda mais a consulta, como é o caso do nome da campanha e do ano de aquisição da informação conforme as **Figura 4.24 a)** e **Figura 4.24 b)**. O procedimento de conexão à BDG, em linguagem de programação PHP (apresentado no Anexo C), é semelhante ao já referido para o *Mapfile* e para o sistema de autenticação, no respeitante à *query* utilizada, que é efetuada com base no operador *UNION*, por via do seguinte código em SQL:

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2;
```

Queries

Cobertura geográfica

X1:

Y1:

X2:

Y2:

(X1,Y1) - coordenadas do canto inferior esquerdo
(X2,Y2) - coordenadas do canto superior direito

Atributos

Campanha:

Ano:

a)

x	y	nome	ambiente
-59451.770	-193246.070	GD Praia	Praia
-60539.519	-194245.745	GS Plataforma	Plataforma
-59440.706	-193206.046	Sondas	Praia
-59451.450	-193222.440	GD Praia	Praia
-59448.870	-193237.410	GD Praia	Praia
-59443.060	-193203.530	GD Praia	Praia
-59456.050	-193244.160	GD Praia	Praia
-61051.468	-193509.753	GD Plataforma	Plataforma
-60704.518	-194842.366	GD Plataforma	Plataforma
-61009.642	-193509.297	GD Plataforma	Plataforma
-59440.341	-193214.781	Sondas	Praia
-59446.080	-193252.880	GD Praia	Praia
-61044.402	-193516.819	GD Plataforma	Plataforma
-61012.036	-193506.562	GD Plataforma	Plataforma
-59453.240	-193235.930	GD Praia	Praia
-61556.986	-194251.838	GD Plataforma	Plataforma
-59440.057	-193203.304	Sondas	Praia

b)

Figura 4.24 — Query: a) Introdução da informação; b) Excerto do resultado.

Para criar uma aplicação dinâmica destinada a um utilizador específico, de entre os *plugins* existentes para explorar e manipular, foram escolhidos os que se apresentam na **Figura 4.25**. Estes *plugins* estão armazenados em `C:\ms4w\apps\pMapper\pMapper-4.4.0\plugins` e foram inseridos no ficheiro `config_default.xml`, cuja descrição se apresenta a seguir.

```
<plugins>export</plugins>
<plugins>scalebar</plugins>
<plugins>transparency</plugins>
<plugins>proj4js</plugins>
<plugins>locateXY</plugins>
<plugins>coordinates</plugins>
<plugins>common</plugins>
<plugins>queryeditor</plugins>
<plugins>unitAndProj</plugins>
```

Figura 4.25 — *Plugins* utilizados na aplicação *p.Mapper*.

- **export** — permite exportar resultados de consulta para os formatos CSV, PDF e *Shapefile*;
- **transparency** — facilita o trabalho do utilizador quando este sobrepõe duas camadas de informação, por exemplo, uma em formato vetorial e outra em formato *raster*. Dá a possibilidade de colocar a camada inferior com transparência, para auxiliar na visualização e exploração da informação espacial da camada superior (**Figura 4.26**);

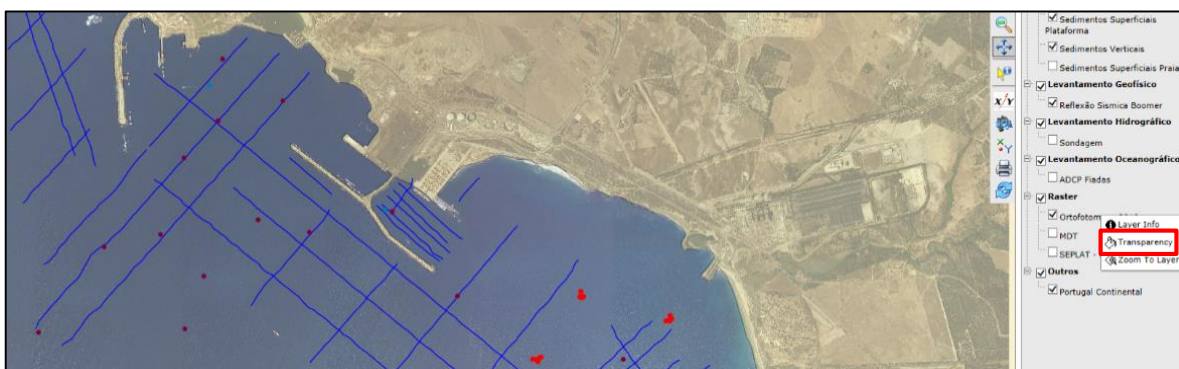


Figura 4.26 — Transparência da *layer*.

- **proj4js** — a biblioteca *proj4* é definida na versão para *JavaScript* pelo que, no caso em estudo, foram estabelecidas as várias projeções usadas no IH em Portugal Continental, Madeira e Açores, o que originou quatro ficheiros, um para cada projeção, como se mostra a seguir e que estão alocados em `C:\ms4w\apps\pMapper\pMapper-4.4.0\plugins\proj4js\defs`.

```
// Portugal PT-TM06
Proj4js.defs["EPSG:3763"] = "+title=PT-TM06 +proj=tmerc +lat_0=39.66825833333333 +lon_0=-8.133108333333334 +k=1 +x_0=0 +y_0=0 +ellps=GRS80 +units=m +no_defs";

// WGS84 UTM 29N
Proj4js.defs["EPSG:32629"] = "+proj=utm +zone=29 +ellps=WGS84 +datum=WGS84 +units=m +no_defs";

// Açores (ITRF93/ PTRAO8 - UTM zona 26N) - Grupo Central e Oriental do Arquipélago dos Açores
proj4.defs("EPSG:5015", "+proj=utm +zone=26 +ellps=GRS80 +towgs84=0,0,0,0,0,0 +units=m +no_defs");
```

```
// Madeira - Selvagens, Porto Santo PTRAO8 / UTM zone 28N
```

```
proj4.defs("EPSG:5016","+proj=utm +zone=28 +ellps=GRS80 +towgs84=0,0,0,0,0,0 +units=m +no_defs");
```

- **unitAndProj** — exhibe a informação correspondente à projeção cartográfica e às unidades dos dados geoespaciais, que estão representados na seguinte interface gráfica (**Figura 4.27**);
- **locateXY** — indica a posição geográfica do ponto para o qual são inseridas as coordenadas.

Na barra de ferramentas, clica-se no botão **X/Y**, para digitar as coordenadas X e Y e poder navegar até à posição indicada. A inserção das coordenadas poderá ser feita nas projeções introduzidas no item anterior “proj4js” (**Figura 4.28 a**) e **Figura 4.28 b**)).

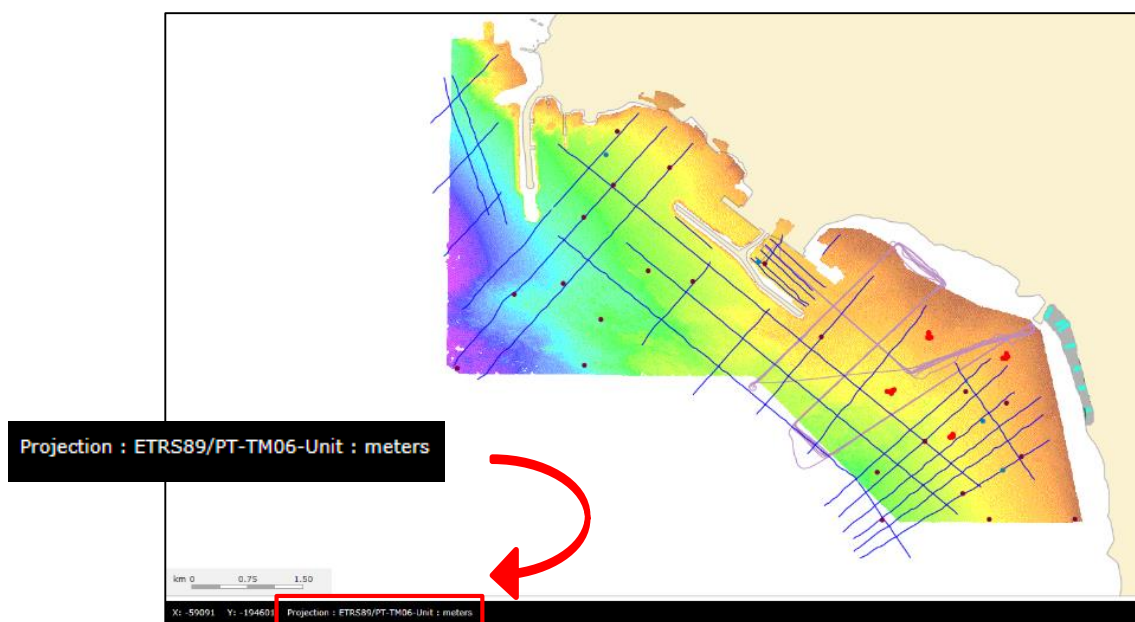


Figura 4.27 — Projeção utilizada na aplicação.

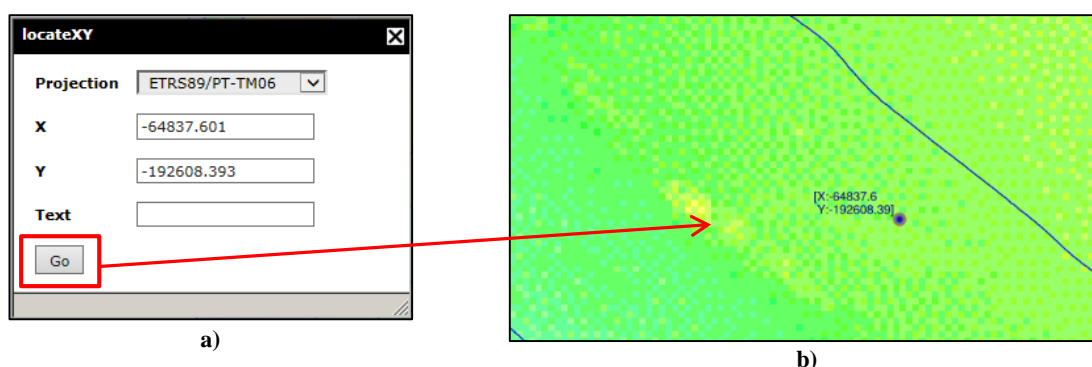



Figura 4.28 — Exemplo de aplicabilidade da ferramenta *locate XY*: a) inserção das coordenadas e b) resultado final.

O Portal proporciona aos utilizadores a interação com as várias camadas de informação espacial existentes na aplicação, o que significa que permite a integração de geoinformação com informação alfanumérica para análise e representação. A vantagem de haver interatividade, entre os utilizadores e a aplicação, possibilita aos mesmos sobrepor níveis de informação, em consonância com o seu objetivo de pesquisa, bem como aceder aos dados alfanuméricos consultando a base de dados, ou seja, através de *queries*.

- **queryeditor** — seleciona *features* de um nível de informação, através de uma *query* por atributos, o que quer dizer que a seleção é efetuada por meio de uma expressão em SQL. Para a concretização desta *query*, clica-se no botão , na barra de ferramentas. O resultado obtido pode ser exportado para os formatos CSV, PDF e *Shapefile* (**Figura 4.29 a**), **Figura 4.29 b**) e **Figura 4.29 c**)).

A **Figura 4.29 a**) mostra o exemplo de uma *query* que foi criada para dar resposta à seguinte questão:

- ✓ Quantas amostras de água em ambiente sedimentar, referentes à plataforma continental, se encontram no patamar correspondente ao fundo?

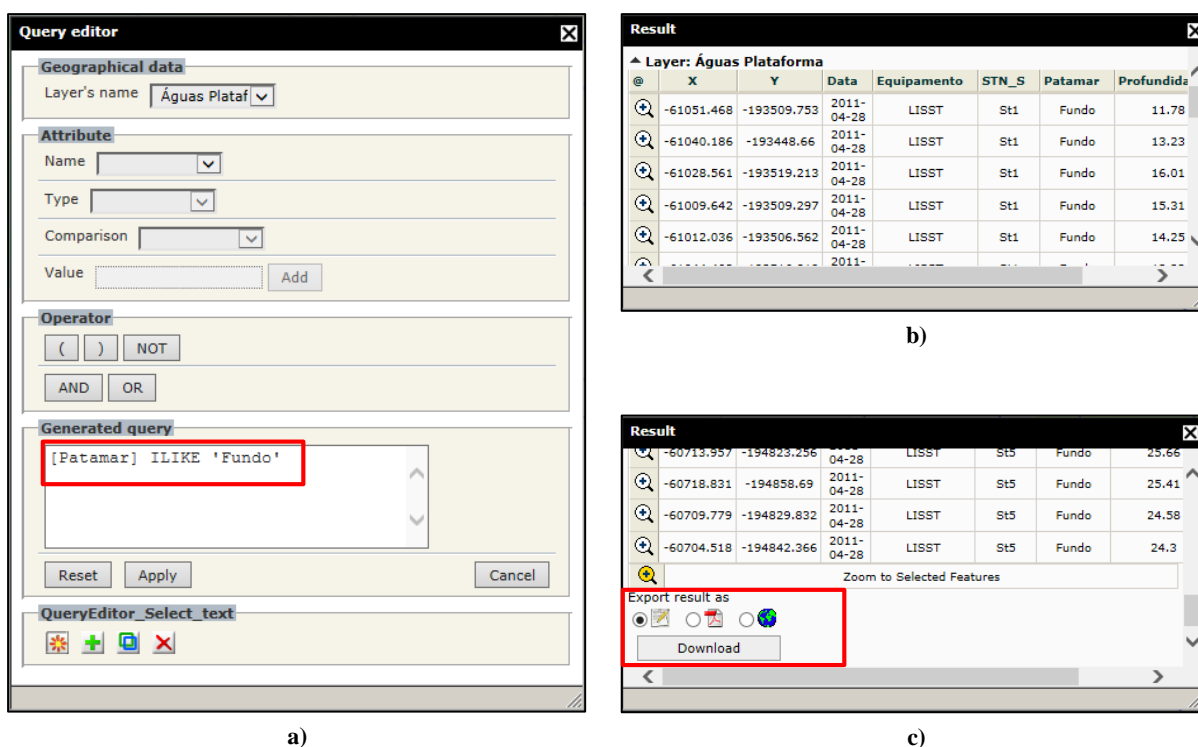



Figura 4.29 — *Query*: a) *Query* em SQL; b) Resultados; c) Formatos para exportar a informação.

A ferramenta *download*  é de grande utilidade dado que, ao clicar neste ícone, o utilizador pode ficar com o mapa temático, que estiver a ser visualizado no momento.

4.2.3 Discussão

Na conceção desta aplicação, foram usadas duas ferramentas, que têm a função de permitir a consulta dos metadados, todavia com procedimentos distintos. Enquanto uma delas possibilita a consulta dos metadados referentes a uma entidade geográfica, a outra permite obter os respeitantes às coordenadas X e Y, ao nome da entidade geográfica e ao ambiente sedimentar envolvente, através da inserção das coordenadas da área, da campanha e do ano. Considera-se que estas duas ferramentas são uma mais-valia, uma vez que dão autonomia ao utilizador para explorar a base de dados, por meio de consultas em SQL de uma forma rápida e intuitiva, o que, consequentemente, permite gerar novos mapas e extrair a informação nos formatos CSV, PDF e/ou *Shapefile*.

Comparando as ferramentas e funcionalidades, que compõem os vários portais apresentados na **Tabela 1.1** e o produto final alcançado neste estudo, verifica-se que o Portal de dados marinhos

conseguiu atingir o objetivo inicialmente proposto, tendo criado as ferramentas de navegação, de interrogação, de pesquisa, de localização e de impressão, tal como as funcionalidades correspondentes ao mapa de referência, às coordenadas X e Y e ao sistema de projeção. Contrariamente a esta evidência, estão os outros portais, em cuja implementação não foram tidos em conta a globalidade destes elementos. Contata-se que todos os portais têm as ferramentas de navegação como o *zoom in*, *zoom out*, *zoom full extent* e o *pan* personalizadas, seguidas das de impressão, de interrogação, de localização e, por último, as de pesquisa. Chama-se a atenção para o facto de apenas o Portal do LNEG e o resultante deste projeto permitirem consultas à base de dados, o que não é um aspeto relevante para os outros portais que foram objeto de pesquisa. No que respeita às funcionalidades, o Portal do SNIMaR é o único que não tem qualquer uma destas opções e somente o projeto internacional *OneGeology*, o do LNEG e o Portal em estudo contemplam o mapa de referência.

Nesta amostra de portais, conclui-se que, entre as opções apresentadas na referida tabela, são consideradas essenciais as ferramentas de navegação, de interrogação e de impressão e que, em relação às funcionalidades, as coordenadas X e Y, assim como a indicação do sistema de projeção são básicas.

Tal como o Portal de dados marinhos, o do município de Arraiolos tem também todas as ferramentas e funcionalidades. Quanto ao SNIG, apresenta ferramentas de navegação, pesquisa e impressão e tem todas as funcionalidades.

CAPÍTULO 5 – CONSIDERAÇÕES FINAIS

Este projeto teve como objetivo o desenvolvimento de um Portal de dados de diferentes origens e formatos que, usando um *software* de código aberto e gratuito, permitisse aos investigadores o acesso a informação, bem como a exploração da mesma, através do uso interativo de um conjunto de ferramentas desenvolvidas para o efeito.

Estando integrada na divisão de Geologia Marinha do Instituto Hidrográfico, a signatária utilizou um projeto multidisciplinar, executado em 2012, e desenvolveu o Portal à medida das necessidades daquela equipa que, através de um simples *click* em cima das entidades geográficas, pode ter acesso, de forma intuitiva, à informação adquirida e disponível na base de dados.

A componente prática desta dissertação teve início com a elaboração do modelo de dados geográfico, no qual foram indicadas as entidades geográficas e convencionais a utilizar e foi feita a discriminação dos vários atributos a integrar em cada tabela. A estrutura da base de dados foi dividida em consonância com as três áreas técnico-científicas que entraram no estudo, a geologia marinha, a hidrografia e a oceanografia. O nome dos campos, correspondentes aos metadados, foi atribuído de acordo com cada tipo de levantamento efetuado, embora ainda tenham sido criados outros de forma a alargar o leque de pesquisa para o utilizador.

A base de dados geográfica *PostgreSQL/PostGIS* mostrou estar capacitada com funções que permitiram efetuar diversas experiências, visando inserir automaticamente os metadados numa tabela ao invés do método manual, que seria mais moroso. Ressalva-se que este processo foi efetuado apenas para as entidades geográficas do tipo ponto, enquanto que as do tipo linha, por opção, foram harmonizadas e inseridas diretamente do formato *shapefile*, uma vez que, em tempo útil, não foi possível explorar o procedimento a efetuar de modo automático. Salienta-se ainda que o modelo de dados, a harmonização e a inserção dessa informação na base de dados foram fases cruciais para atingir o objetivo da implementação da base de dados.

A viabilidade de realizar consultas de informação, por meio de *queries* espaciais, é bastante significativa por permitir que o resultado obtido possa ser observado num ambiente de sistemas de informação geográfica, de código aberto e gratuito ou proprietário, neste caso específico, testado e validado na solução de código aberto *QGIS Desktop 2.14.3*. É importante realçar este facto, porque permite aferir que existe compatibilidade entre a base de dados geográfica criada e o *software* de código aberto, no âmbito dos SIGs.

Os programas utilizados no desenrolar deste estudo, designadamente, o *PostgreSQL/PostGIS* e o *MapServer* são acessíveis, mesmo para quem está a iniciar este processo de aprendizagem. Qualquer um deles tem manuais intuitivos, que permitem um nível de formação acessível e dinâmico.

Relativamente ao servidor de mapas *MapServer*, é fundamental referir que o *Mapfile* teve um contributo essencial na conceção da aplicação *Web*. É neste ficheiro que está contida toda a informação considerada relevante e que foi utilizada para personalizar a plataforma, de forma evidente e com possibilidade de interação com o utilizador. A sua importância prende-se também com a capacidade que tem de efetuar remotamente a ligação à base de dados espacial, indicando os seus parâmetros de configuração e com uma *query* em que é extraída a informação, através do campo correspondente à geometria. Existe igualmente compatibilidade entre a BDG utilizada e o servidor de mapas, o que é considerado bastante satisfatório. Para o IH, em particular para a GM, a importância da interoperabilidade da informação geográfica destas soluções prende-se, prende-se com o facto de se considerar relevante implementar uma base de dados, que seja compatível com todos ou com a maior parte deles, por acrescentar a mais-valia de não haver redundância de informação e de ser de fácil

visualização.

Efetivamente, nas aplicações criadas neste projeto, designadamente, o sistema de autenticação e a realização de *queries*, através da inserção das coordenadas da área envolvente, a conexão à base de dados espacial foi feita diretamente nos *scripts* construídos em linguagem de programação PHP, não havendo ligação ao ficheiro *Mapfile*. Esta realidade mostra que existem duas maneiras distintas de aceder à BDG, de uma forma eficaz e fiável, garantindo que a sua manipulação seja realizada remotamente.

Constatou-se que a junção destes três produtos, *PostgreSQL/PostGIS*, *MapServer* e *p.Mapper*, mostra ter boa interoperabilidade, o que significa que, à medida que se vai adicionando novas entidades geográficas à base de dados espacial, ou manipulando as existentes, é possível visualizar tanto as atualizações como as alterações na interface gráfica do utilizador. Verifica-se ainda que, com esta fusão, é exequível a construção de uma aplicação acessível e intuitiva para o utilizador, com uma interface gráfica atrativa e, acima de tudo, funcional.

O Portal de dados em ambiente marinho, criado como protótipo, pode ser considerado como um gestor de informação geográfica, visando a possível centralização e organização dos vários tipos de informação. O facto dos objetivos iniciais de desenvolvimento de um portal de dados terem sido atingidos mostra que o *software* de código aberto e gratuito também está operacional, por analogia com os programas proprietários, ao prover a disponibilização de informação remotamente. Tal evidência prova que é possível reduzir na despesa com aquisição de *soluções* proprietários, havendo, à partida, outras soluções de código aberto e gratuitas a considerar.

Esta aplicação é a base que fica ao dispor de um conjunto de utilizadores, podendo ser melhorada e otimizada consoante as suas necessidades, visto permitir que, em qualquer altura, possa ser alterada mediante a introdução de novas funcionalidades.

A par do que foi referido no parágrafo anterior e tendo em conta a quantidade de informação existente, ficam criadas as condições para dar continuidade a este projeto. Neste sentido, em trabalhos futuros, a base de dados pode ir sendo alimentada com informação harmonizada de outras campanhas, podendo também criar-se a indexação espacial para cada uma das entidades geográficas, visando a otimização das consultas em SQL.

REFERÊNCIAS BIBLIOGRÁFICAS

- Akbari, M. and S.R.H. Peikar (2014) “Evaluation of free/open source software using OSMM model case study: WebGIS and spatial database.” *Adv. Comput. Sci. anInt. J.*, Vol. 3, No. 5, pp. 34–43.
- APACHE (2017) “The Apache software foundation” <https://www.apache.org/licenses/GPL-compatibility.html> (acedido em 01 de setembro de 2017).
- APDSI (2004) Open Source Software – Que oportunidades em Portugal? Associação para a Promoção e Desenvolvimento da Sociedade da Informação. Lisboa. Portugal. 46 p. Disponível online no endereço URL: http://www.algebraica.pt/i_ap/bo2/data/upimages/Estudo_Open_Source_com_capa.pdf (acedido em 20 de julho de 2017).
- Bonnici, A.M. (2005) *Web GIS Software Comparison Framework*. Sir Sandford Fleming College. Disponível online no endereço URL: <https://www.scribd.com/document/213970393/Webgis-Framework> (acedido em 09 de agosto de 2017).
- Borges, K. (1997) *Modelagem de dados geográficos: Uma extensão do modelo OMT para aplicações geográficas*. Escola de Governo-Fundação João Pinheiro, Belo Horizonte, Brasil, 139p..
- Borges, K.A.V., C.A. Davis Jr. and A.H.F. Laender (2001) “OMT-G: An Object-Oriented Data Model for Geographic Applications.” *GeoInformatica* 5:3, pp. 221-260.
- Borges, K.A., C.A. Davis Jr. and A. Laender (2005) “Bancos de Dados Geográficos – Modelagem conceitual de dados geográficos.” Disponível online no endereço URL: <http://www.dpi.inpe.br/livros/bdados/cap3.pdf> (acedido em 14 de agosto de 2017).
- Bughi, C.H. (2007) *Mini-curso MapServer Avançado PHP/MapScript*. Universidade do Vale do Itajaí, Brasil, 74p..
- Butler, H., M. Daly, A. Doyle, S. Gillies, T. Schaub, and S. Hagen (2016) The GeoJSON Format Disponível online no endereço URL: <https://tools.ietf.org/html/draft-ietf-geojson-02> (acedido em 20 de setembro de 2017).
- Cabral, P.S. (2008) *Novas ferramentas para monitoramento ambiental usando sig web*. Tese de Doutorado, Natal/RN, Universidade Federal do Rio Grande do Norte, Brasil.
- Caçador, F. (2008) Protocolo da ESOP com AMA dinamiza open source na Administração Pública. Disponível online no endereço URL: <http://tek.sapo.pt/noticias/computadores/artigos/protocolo-da-esop-com-ama-dinamiza-open-source-na-administracao-publica> (acedido em 02 de agosto de 2017).
- Câmara, G. (2005) *Bancos de Dados Geográficos – Representação Computacional de Dados geográficos*. Disponível online no endereço URL: <http://www.dpi.inpe.br/livros/bdados/cap1.pdf> (acedido em 15 de agosto de 2017).
- Catalão, J. (2015). Apontamentos sobre a unidade curricular Sistemas de Localização e Geoinformação. Texto não publicado, Universidade de Lisboa, Lisboa.
- Charneca, N.M.P. (2013) *Modelação de dados geográficos aplicada ao planeamento e gestão de recursos hídricos*. Tese de Doutorado, Faculdade de Ciências da Universidade de Lisboa, Lisboa, 268p.
- Condeça, J.J.V. (2009). *Aplicações SIG no apoio aos processos de licenciamento na ARH do Alentejo*,

- I.P. *O Caso das captações de água*. Dissertação de Mestrado, Instituto Superior de Estatística e Gestão de Informação, Universidade Nova de Lisboa, Lisboa, 110p.
- CHRIS-DURAN (2014) “NOOB WEB DEV – A Rookie’s Experience with Web Development.” Disponível online no endereço URL: <http://chris-duran.com/?m=201412> (acedido em 01 de agosto de 2017).
- ESOP (2017) Disponível online no endereço URL: <http://www.esop.pt/Default/pt/Homepage> (acedido em 01 de setembro de 2017).
- Ferreira, A.A., D.B. Barreiros, and I.M.B.R. Barbosa (2015) “SIGWeb para Dados de Qualidade de Água em Bacias Hidrográficas.” *Revista CIENTEC*, Vol. 7, No. 1, pp. 43–55.
- Freimuth, P., A. Christl, and H. Tveite (2017) Cartographical Symbol Construction with MapServer. Disponível online no endereço URL: <http://mapserver.org/mapfile/symbology/construction.html> (acedido em 31 de agosto de 2017).
- EPSG (2017) “International Association of Oil & Gas Producers.” Disponível online no endereço URL: <http://www.epsg.org> (acedido em 02 de agosto de 2017).
- FSF (2017) “The Free Software Definition.” Disponível online no endereço URL: <http://www.gnu.org/philosophy/free-sw.html> (acedido em 01 de agosto de 2017).
- GDAL (2017) “GDAL - Biblioteca de Abstrações de Dados Geo-Espaciais.” Disponível online no endereço URL: http://www.gdal.org/index_br.html (acedido em 09 de agosto de 2017).
- Godinho, R.J.N. (2012) *O Open Source Software na Administração Pública Central Portuguesa. Grau e tendências de utilização; Fatores inibidores e facilitadores da sua adoção*. Dissertação de Mestrado, Instituto Universitário de Lisboa, Lisboa, 212p.
- Gorni, D., M. Giannotti, A. Knopik, P. Brito, and M. Rodrigues (2007) “Open source web GIS Sistema de Informação Geográfica de Expedições.” In: *Anais XIII Simpósio Brasileiro de Sensoriamento Remoto*, Ed. 21-26 abril, Universidade de São Paulo, São Paulo, Brasil, pp. 3953-3960.
- Hellerstein, J.M., J.F. Naughton, and A. Pfeffer (1995) Generalized search trees for databases systems. In: *international Conference in VLDB*, 21., set., Zurich, Switzerland. Proceeding. San Francisco: Morgan Kaufman, 562-573.
- IH (2017) “Enquadramento Estratégico.” Disponível online no endereço URL: <http://www.hidrografico.pt/enquadramento-estrategico.php> (acedido em 8 de setembro de 2017).
- Ingvarsson, T.M. (2004) “MapServer An OpenSource Web-Map Applications” Disponível online no endereço URL: <http://www.gdmc.nl/publications/2004/OpenSource-MapServer.pdf> (acedido em 17 de Agosto de 2017).
- ISO 19103 (2015) <https://www.iso.org/standard/56734.html> (acedido em 24 de agosto de 2017).
- Kanegae, E. P. (2005) *Curso de Introdução ao MapServer*.
- Kropla, B. (2005) *Mapserver: Open Source GIS Development*. (New York: Springer) Berkeley: Apress.
- Longley, P.A., M.F. Goodchild, D.J. Maguire, and D.W. Rhind (2005) “Geographic information systems and science.” 2nd Edition, John Wiley & Sons Ltd., Barcelona, 536p.
- MapServer (2017) “MapServer 7.0.6 Documentation.” Disponível online no endereço URL: <http://mapserver.org/documentation.html> (acedido em 10 setembro de 2017).

- Maravilhas-Lopes, S.P. (2013) “A importância dos profissionais da gestão da informação para as organizações.” *Biblios: Revista de Bibliotecologia y Ciencias de La Información*, No.51. Disponível online no endereço URL: <https://biblios.pitt.edu/ojs/index.php/biblios/article/view/102> (acedido em 17 de agosto de 2017).
- McKenna, J., Mitchell, T. and Nacionales, P. (2007) “Getting Started with MapServer, Part 1.” Open Source Geospatial Foundation, 65p.
- Melo, R., R. Santos, and A. Rodrigues (2016) “Recuperação dos registos de reflexão sísmica - modo analógico para o digital com a máquina fotográfica.” *Actas 4^{as} Jornadas de Engenharia Hidrográfica*, Lisboa pp. 385-388.
- Muilwijk, R. (2016) Disponível online no endereço URL: <https://opensource.com/business/16/8/top-5-open-source-web-servers> (acedido em 01 de setembro de 2017).
- Neto, A.P. (2003) *PostgreSQL – Técnicas Avançadas Versões Open Source 7.x*. Érica, II Série, São Paulo, Brasil.
- Oliveira, P. (2016) Disponível online no endereço URL: <https://www.escolalinux.com.br/blog/servidor-web-o-que-e-e-como-escolher-um-para-seu-site> (acedido em 08 de agosto de 2017).
- Oliviero, C.A.J. (2001) *Faça um Site JavaScript Orientado por Projeto*. Érica, 4^a Edição, São Paulo, Brasil.
- OGC (2010) Implementation Standard for Geographic information – Simple feature access – Part2: SQL option. OGC 06-104r4, version 1.2.1. 111 p..
- OGC (2017) “Open GeoSpatial Consortium.” Disponível online no endereço URL: <http://www.opengeospatial.org> (acedido em 14 de agosto de 2017).
- OPENSOURCE (2007) “Open Source Initiative” Disponível online no endereço URL: <https://opensource.org/osd> (acedido em 12 de agosto de 2017).
- OPENSOURCE (2012) “Open Source Initiative” Disponível online no endereço URL: <https://opensource.org/history> (acedido em 12 de agosto de 2017).
- Peng, Z.R. and M.H. Tsou (2003) *Internet GIS - Distributed geographic information services for the internet and Wireless Networks*. 1st Edition, John Wiley & Sons Inc., New Jersey, 679p.
- PHP (2017) PHP Home Page. Disponível online no endereço URL: <http://php.net/> (acedido em 25 de julho de 2017).
- Pimenta, F.M., E.C. Landau, A. Hirsch, and D.P. Guimaraes (2012) “Servidores de mapas: programação para disponibilizar dados geográficos multidisciplinares utilizando tecnologias livres.” Brasília, DF: Embrapa, 216p.
- Pimenta, F.M., E.C. Landau, A. Hirsch, and D.P. Guimaraes (2013) “Programação de Servidor de Mapas para disponibilização de dados geográficos multidisciplinares utilizando tecnologias livres.” Sete Lagoas: Embrapa Milho e Sorgo, 45p.: il. -- (Documentos / Embrapa Milho e Sorgo, ISSN 1518-4277; 158).
- PMAPPER (2017) “p.mapper - a MapServer PHP/MapScript Framework.” Disponível online no endereço URL: <http://www.pmapper.net> (acedido em 21 de agosto de 2017).
- PostGIS (2017) PostGIS Home Page. Disponível online no endereço URL: <http://postgis.refractory.net/> (acedido em 21 de agosto de 2017).

- PostgreSQL (2017) PostgreSQL Home Page. Disponível online no endereço URL: <http://www.postgresql.org/> (acedido em 20 de agosto de 2017).
- PROJ.4 (2017) “Welcome to PROJ.4.” Disponível online no endereço URL: <http://trac.osgeo.org/proj/> (acedido em 15 de agosto de 2017).
- Queiroz, G.R. and K.R. Ferreira (2005) *Bancos de Dados Geográficos – Modelagem conceitual de dados geográficos*. Disponível online no endereço URL: <http://www.dpi.inpe.br/livros/bdados/cap8.pdf> (acedido em 14 de agosto de 2017).
- Rodrigues, A., A. Oliveira, and A.I. Santos (2012) “Caraterização ambiental da área de expansão marítima do porto de Sines e região envolvente.” Relatório técnico final, Texto não publicado, Instituto Hidrográfico, 1515p.
- Singh, P.S., C. Dibyajyoti and S. Sudhakar (2012) “Development of a Web Based GIS Application for Spatial Natural Resources Information System Using Effective Open Source Software and Standards.” *Journal of Geographic Information System*, No. 4, pp. 261-266.
- Steiniger, S. and A.J.Hunter (2012) “Free and Open Source GIS Software for Building a Spatial Data Infrastructure,” In: E. Bocher and M. Neteler, Eds., *Geospatial Free and Open Source Software in the 21st Century*, Springer-Verlag Berlin Heidelberg, pp. 247-261.
- SOFTWARELIVRE (2017) Disponível online no endereço URL: <http://softwarelivre.org/php/servidor-web-apache> (acedido em 08 de agosto de 2017).
- Stonebraker, M., L.A. Rowe, and M. Hirohama (1990) “The implementation of postgres.” University of California, Berkeley, 36p.
- Trezentos, P. and A. Simão (2004) “Software aberto na administração pública.” 1ª Edição, Edição UMIC, Lisboa, 78p.
- Vidigal, L. (2014) “Gestão da Informação e conhecimento na AP.” Disponível online no endereço URL: <https://pt.slideshare.net/vidigal/gestao-da-informacao-luis-vidigal-sociedade-da-informacao> (acedido em 07 de agosto de 2017).

ANEXOS

Anexo A — Criação das tabelas em SQL

```
CREATE TABLE campanha
(
  nome_camp varchar(50) NOT NULL,
  data_inicio date,
  data_fim date,
  ano numeric(4) NOT NULL,
  local varchar(15),
  navio varchar(15),
  resp_tecnico varchar(20),
  resp_missao varchar(20),
  CONSTRAINT campanha_pkey PRIMARY KEY (nome_camp)
);
```

```
CREATE TABLE gs_plataforma
(
  gid_gsplt serial NOT NULL,
  nome_camp varchar(50) NOT NULL,
  nome varchar(20) NOT NULL,
  ambiente_sed varchar(10) NOT NULL,
  equipamento varchar(50),
  data date NOT NULL,
  ano numeric(4) NOT NULL,
  stn varchar(4),
  x numeric(9, 3),
  y numeric(9, 3),
  z_zh numeric(5, 3),
  geom geometry,
  datum_orig varchar(15),
  CONSTRAINT gs_plataforma_pkey PRIMARY KEY (gid_gsplt),
  CONSTRAINT enforce_dims_geom CHECK (st_ndims (geom) = 2),
  CONSTRAINT enforce_geotype_geom CHECK (geometrytype (geom) = 'MULTIPOINT'::text OR geom IS NULL),
  CONSTRAINT enforce_srid_geom CHECK (st_srid (geom) = 3763),
  FOREIGN KEY (nome_camp) references campanha (nome_camp)
);
```

```
CREATE TABLE gs_praia
(
  gid_gspr serial NOT NULL,
  nome_camp varchar(50) NOT NULL,
  nome varchar(10) NOT NULL,
  ambiente_sed varchar(5) NOT NULL,
  data date NOT NULL,
  ano numeric(4) NOT NULL,
  equipamento varchar(50),
  perfil varchar(10),
  stn varchar(4),
  x numeric(9, 3),
  y numeric(9, 3),
  z_zh numeric(5, 3),
```

```
geom geometry,  
datum_orig varchar(15),  
CONSTRAINT gs_praia_pkey PRIMARY KEY (gid_gspr),  
CONSTRAINT enforce_dims_geom CHECK (st_ndims (geom) = 2),  
CONSTRAINT enforce_geotype_geom CHECK (geometrytype (geom) = 'MULTIPOINT'::text OR geom IS  
NULL),  
CONSTRAINT enforce_srid_geom CHECK (st_srid (geom) = 3763),  
FOREIGN KEY (nome_camp) references campanha (nome_camp)  
);
```

```
CREATE TABLE gs_vertical  
(  
gid_gsv serial NOT NULL,  
nome_camp varchar(50) NOT NULL,  
nome varchar(10) NOT NULL,  
ambiente_sed varchar(10) NOT NULL,  
data date NOT NULL,  
ano numeric(4) NOT NULL,  
equipamento varchar(50),  
x numeric(9, 3),  
y numeric(9, 3),  
fora time,  
espessura numeric(3,2),  
stn character(3),  
z_zh numeric (5, 3),  
geom geometry,  
datum_orig varchar(15),  
CONSTRAINT gs_vertical_pkey PRIMARY KEY (gid_gsv),  
CONSTRAINT enforce_dims_geom CHECK (st_ndims (geom) = 2),  
CONSTRAINT enforce_geotype_geom CHECK (geometrytype (geom) = 'MULTIPOINT'::text OR geom IS  
NULL),  
CONSTRAINT enforce_srid_geom CHECK (st_srid (geom) = 3763),  
FOREIGN KEY (nome_camp) references campanha (nome_camp)  
);
```

```
CREATE TABLE gd_praia  
(  
gid_gdpr serial NOT NULL,  
nome_camp varchar(50) NOT NULL,  
nome varchar(10) NOT NULL,  
ambiente_sed varchar(5) NOT NULL,  
data date NOT NULL,  
ano numeric(4) NOT NULL,  
equipamento varchar(30),  
x numeric(9, 3),  
y numeric(9, 3),  
z_zh numeric(5, 3),  
menos_zh numeric(5, 3),  
geom geometry,  
datum_orig varchar(15),  
CONSTRAINT gd_praia_pkey PRIMARY KEY (gid_gdpr),  
CONSTRAINT enforce_dims_geom CHECK (st_ndims (geom) = 2),  
CONSTRAINT enforce_geotype_geom CHECK (geometrytype (geom) = 'MULTIPOINT'::text OR geom IS
```

```
NULL),  
CONSTRAINT enforce_srid_geom CHECK (st_srid(geom) = 3763),  
FOREIGN KEY (nome_camp) references campanha (nome_camp)  
);
```

```
CREATE TABLE gd_plataforma
```

```
(  
    gid_gdplt serial NOT NULL,  
    nome_camp varchar(50) NOT NULL,  
    nome varchar(20) NOT NULL,  
    ambiente_sed varchar(10) NOT NULL,  
    data date NOT NULL,  
    ano numeric(4) NOT NULL,  
    equipamento varchar(30),  
    stn_s varchar(5),  
    ciclo_c varchar(3),  
    patamar_p varchar(12),  
    x numeric(9, 3),  
    y numeric(9, 3),  
    hh_mm_inic time,  
    hh_mm_fim time,  
    prof_lisst numeric(4, 2),  
    conc_lisst numeric(4, 2),  
    z_zh numeric (5, 3),  
    geom geometry,  
    datum_orig varchar(15),  
    CONSTRAINT gd_plataforma_pkey PRIMARY KEY (gid_gdplt),  
    CONSTRAINT enforce_dims_geom CHECK (st_ndims (geom) = 2),  
    CONSTRAINT enforce_geotype_geom CHECK (geometrytype (geom) = 'MULTIPOINT'::text OR geom IS  
    NULL),  
    CONSTRAINT enforce_srid_geom CHECK (st_srid (geom) = 3763),  
    FOREIGN KEY (nome_camp) references campanha (nome_camp)  
);
```

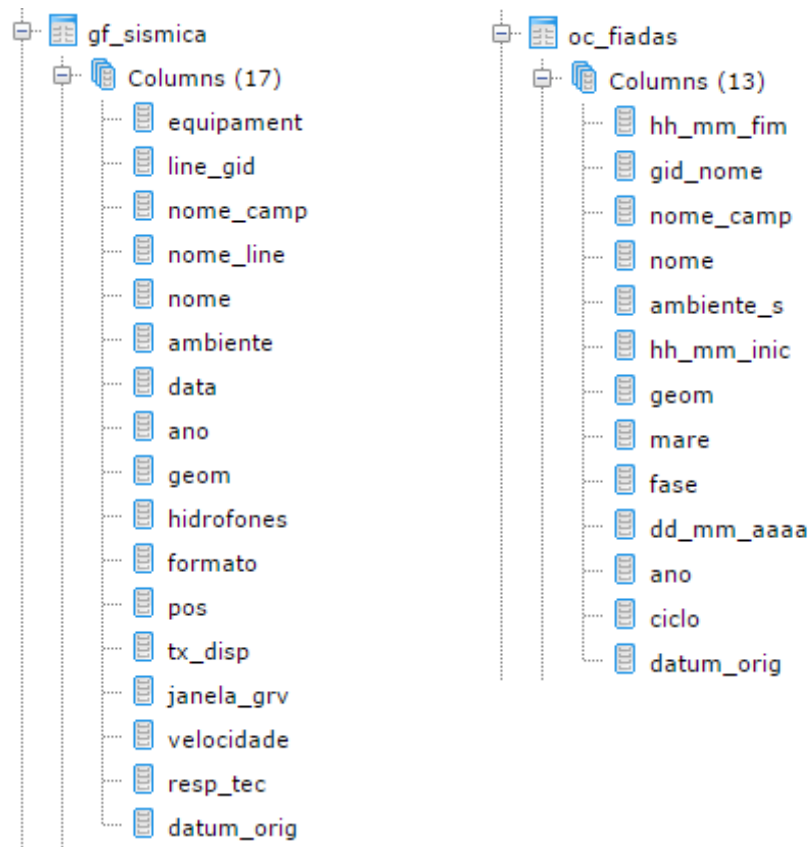
```
CREATE TABLE hi_sondagem
```

```
(  
    gid_hi integer NOT NULL,  
    nome_camp varchar(50) NOT NULL,  
    nome varchar(10) NOT NULL,  
    ambiente_sed varchar(5) NOT NULL,  
    data date NOT NULL,  
    ano numeric(4) NOT NULL,  
    equipamento varchar(50),  
    x numeric(9, 3),  
    y numeric(9, 3),  
    z_zh numeric (5, 3),  
    menosz_zh numeric (5, 3),  
    geom geometry,  
    datum_orig varchar(15),  
    CONSTRAINT hi_sondas_pkey PRIMARY KEY (gid_hi),  
    CONSTRAINT enforce_dims_geom CHECK (st_ndims (geom) = 2),  
    CONSTRAINT enforce_geotype_geom CHECK (geometrytype (geom) = 'MULTIPOINT'::text OR geom IS  
    NULL),
```

```
CONSTRAINT enforce_srid_geom CHECK (st_srid (geom) = 3763),  
FOREIGN KEY (nome_camp) references campanha (nome_camp)  
);
```

```
CREATE TABLE users  
(  
    userid integer NOT NULL,  
    nome varchar(50),  
    username varchar (20) NOT NULL,  
    password varchar (20) NOT NULL,  
    isadmin boolean,  
    CONSTRAINT users_pkey PRIMARY KEY (userid)  
);
```

Campos criados para as tabelas correspondentes às entidades *gf_sismica* e *oc_fiadas*, que foram inseridas diretamente da *shapefile*.



Anexo B — Ficheiro *Mapfile*

```
#
# Start of map file
#
MAP

    EXTENT -67576.581 -197000.000 -58363.104 -188000.000
    UNITS meters
    SIZE 600 500
    SHAPEPATH "../..../pmapper_demodata"
    SYMBOLSET "../common/symbols/symbols-pmapper.sym"
    FONTSET "../common/fonts/msfontset.txt"
    RESOLUTION 96
    IMAGETYPE png
    INTERLACE OFF
    PROJECTION
    "init=epsg:3763"
    END

    #
    # Start of web interface definition
    #
    WEB
    TEMPLATE "map.html"
    IMAGEPATH "/ms4w/tmp/ms_tmp/"
    IMAGEURL "/ms_tmp/"
    METADATA
    END # Metadata
    END # Web

    #
    # Start of Reference map definition
    #
    REFERENCE
    EXTENT -67576.581 -197000.000 -58363.104 -188000.000
    IMAGE "../..../images/reference.png"
    SIZE 186 153
    COLOR -1 -1 -1
    OUTLINECOLOR 255 0 0
    END # Reference

    #
    # Start of ScaleBar definition
    #
    SCALEBAR
    STATUS off
    TRANSPARENT off
    INTERVALS 4
    SIZE 200 3
    UNITS meters
    COLOR 250 250 250
```

```
OUTLINECOLOR 0 0 0
BACKGROUNDCOLOR 100 100 100
STYLE 0
POSTLABELCACHE true
LABEL
COLOR 0 0 90
#OUTLINECOLOR 200 200 200
SIZE small
END # Label
END # ScaleBar
```

```
# SYMBOLS USED IN PMAPPER
# - 'circle' always necessary (used e.g. for highlight)
# - 'square' used in current map file
# Symbols can also be defined via tag SYMBOLSET (see above)
Symbol
Name 'circle'
Type ELLIPSE
Filled TRUE
Points
  1 1
END
END
```

```
Symbol
Name 'square'
Type VECTOR
Filled TRUE
Points
  0 1
  0 0
  1 0
  1 1
  0 1
END
END
```

```
#===== START OF LAYER SECTION =====#
```

```
#----- RASTER -----#
```

```
LAYER
  NAME mdt
  DATA "/ms4w/apps/pmapper/pmapper-4.4.0/images/mdt1.gif"
  STATUS ON
  TYPE RASTER
  OFFSITE 70 74 66
  PROJECTION
    "init=epsg:3763"
  END
  METADATA
    "DESCRIPTION" "MDT"
```


END
END

LAYER
NAME orto2010
DATA "/ms4w/apps/pmapper/pmapper-4.4.0/images/orto2010.png"
STATUS ON
TYPE RASTER
OFFSITE 70 74 66
PROJECTION
"init=epsg:3763"
END
METADATA
"DESCRIPTION" "Ortofotomapa 2010"
END
END

LAYER
NAME sed6
DATA "/ms4w/apps/pmapper/pmapper-4.4.0/images/sed6.gif"
STATUS ON
TYPE RASTER
OFFSITE 70 74 66
PROJECTION
"init=epsg:3763"
END
METADATA
"DESCRIPTION" "SEPLAT - SED6"
END
END

LAYER#-----Portugal Continental
NAME "portugal"
TYPE polygon
DATA "PT_ETRS89TM06PT"
TRANSPARENCY 30
TEMPLATE void
PROJECTION
"init=epsg:3763"
END
METADATA
"DESCRIPTION" "Portugal Continental"
END # Metadata
CLASS
Name 'Portugal Continental'
COLOR 242 215 106
OUTLINECOLOR 0 0 0
END # Class
END # Layer

```
#
# Start of Layer Sondas
#
LAYER
  NAME "sondas"
  TYPE point
  CONNECTIONTYPE POSTGIS
  PROCESSING "CLOSE_CONNECTION=DEFER"
  CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
  DATA "geom FROM hi_sondas USING UNIQUE gid_hi USING SRID=3763"
  TOLERANCE 6
  TOLERANCEUNITS pixels
  LABELMAXSCALE 8000000
  TEMPLATE void
  PROJECTION
    "init=epsg:3763"
  END
  METADATA
    "DESCRIPTION" "Sondagem"
    "RESULT_FIELDS" "nome_camp, data, x, y, equipamento, z_zh, menosz_zh"
    "RESULT_HEADERS" "Campanha, Data, X, Y, Equipamento, Z_ZH, MenosZ_ZH"
    "LAYER_ENCODING" "UTF-8"
  END # Metadata
  CLASS
    # Name 'Sondagem'
    SYMBOL 'circle'
    SIZE 4
    COLOR 0 128 64
  END # Class
END # Layer
```

```
#
# Start of Layer Fiadas OC
#
LAYER
  NAME "oc_fiadas"
  TYPE line
  CONNECTIONTYPE POSTGIS
  PROCESSING "CLOSE_CONNECTION=DEFER"
  CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
  DATA "geom FROM oc_fiadas USING UNIQUE gid_nome USING SRID=3763"
  TOLERANCE 6
  TOLERANCEUNITS pixels
  TEMPLATE void
  PROJECTION
    "init=epsg:3763"
  END
  METADATA
    "DESCRIPTION" "ADCP Fiadas"
    "RESULT_FIELDS" "nome_camp, data, equipamento"
    "RESULT_HEADERS" "Campanha, Data, Equipamento"
```

```
"LAYER_ENCODING" "UTF-8"
END # Metadata
CLASS
  # Name 'Fiadas ADCP'
  COLOR 190 140 200
END # Class
END # Layer

#
# Start of Layer Sismica
#
LAYER
  NAME "sismica"
  TYPE line
  CONNECTIONTYPE POSTGIS
  PROCESSING "CLOSE_CONNECTION=DEFER"
  CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
  DATA "geom FROM gf_sismica USING UNIQUE line_gid USING SRID=3763"
  TOLERANCE 3
  TOLERANCEUNITS pixels
  TEMPLATE void
  PROJECTION
    "init=epsg:3763"
  END
  METADATA
    "DESCRIPTION" "Reflexão Sismica Boomer"
    "RESULT_FIELDS" "nome_camp, nome_line, equipamento, hidrofonos, formato"
    "RESULT_HEADERS" "Campanha, Fiada, Equipamento, Hidrofonos, Formato"
    "LAYER_ENCODING" "UTF-8"
  END # Metadata
  CLASS
    # Name 'Sismica'
    COLOR 0 0 255
  END # Class
END # Layer

#
# Start of Layer Águas Praia
#
LAYER
  NAME "gd_praia"
  TYPE point
  CONNECTIONTYPE POSTGIS
  PROCESSING "CLOSE_CONNECTION=DEFER"
  CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
  DATA "geom FROM gd_praia USING UNIQUE gid_gdpr USING SRID=3763"
  TOLERANCE 6
  TOLERANCEUNITS pixels
  TEMPLATE void
  PROJECTION
```

```
"init=epsg:3763"
END
METADATA
  "DESCRIPTION" "Águas Praia"
  "RESULT_FIELDS" "nome_camp, x, y, ano, equipamento, z_zh, menosz_zh, datum_orig"
  "RESULT_HEADERS" "Campanha, X, Y, Ano, Equipamento, Z_ZH, MenosZ_ZH,
  Datum_origem"
  "LAYER_ENCODING" "UTF-8"
END # Metadata
CLASS
  #Name 'Dinâmica_Praia'
  SYMBOL 'circle'
  SIZE 5
  COLOR 180 180 180
END # Class
END # Layer

#
# Start of Layer Dinâmica Sedimentar Plataforma
#
LAYER
  NAME "gd_plataforma"
  TYPE point
  CONNECTIONTYPE POSTGIS
  PROCESSING "CLOSE_CONNECTION=DEFER"
  CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
  DATA "geom FROM gd_plataforma USING UNIQUE gid_gdplt USING SRID=3763"
  TOLERANCE 6
  TOLERANCEUNITS pixels
  TEMPLATE void
  PROJECTION
    "init=epsg:3763"
  END
  METADATA
    "DESCRIPTION" "Águas Plataforma"
    "RESULT_FIELDS" "x, y, data, equipamento, stn_s, patamar_p, prof_lisst, z_zh"
    "RESULT_HEADERS" "X, Y, Data, Equipamento, STN_S, Patamar, Profundidade, Z_ZH"
    "LAYER_ENCODING" "UTF-8"
  END # Metadata
  CLASS
    # Name 'Dinâmica_Plataforma'
    SYMBOL 'circle'
    SIZE 5
    COLOR 255 0 0
  END # Class
END # Layer

#
# Start of Layer Amostragem Sedimentos Plataforma
#
LAYER
```

```
NAME "gs_plataforma"
TYPE point
CONNECTIONTYPE POSTGIS
CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
DATA "geom FROM gs_plataforma USING UNIQUE gid_gsplt USING SRID=3763"
TOLERANCE 6
TOLERANCEUNITS pixels
PROJECTION
    "init=epsg:3763"
END
METADATA
    "DESCRIPTION" "Sedimentos Superficiais Plataforma"
    "RESULT_FIELDS" " x, y, data, equipamento, stn, z_zh"
    "RESULT_HEADERS" "X, Y, Data, Equipamento, STN_S, Z_ZH"
    "LAYER_ENCODING" "UTF-8"
END # Metadata
CLASS
    # Name 'Amostragem_Plataforma'
    SYMBOL 'circle'
    SIZE 5
    COLOR    128 0 64
    TEMPLATE void
END # Class
END # Layer
```

```
#
# Start of Layer Amostragem Sedimentos Vertical
#
LAYER
    NAME "gs_vertical"
    TYPE point
    CONNECTIONTYPE POSTGIS
    PROCESSING "CLOSE_CONNECTION=DEFER"
    CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
    DATA "geom FROM gs_vertical USING UNIQUE gid_gsv USING SRID=3763"
    TOLERANCE 6
    TOLERANCEUNITS pixels
    TEMPLATE void
    PROJECTION
        "init=epsg:3763"
    END
    METADATA
        "DESCRIPTION" "Sedimentos Verticais"
        "RESULT_FIELDS" "x, y, data, equipamento, fora, espessura, stn, z_zh"
        "RESULT_HEADERS" "X, Y, Data, Equipamento, Fora, Espessura, STN, Z_ZH"
        "LAYER_ENCODING" "UTF-8"
    END # Metadata
    CLASS
        # Name 'Amostragem_Vertical'
        SYMBOL 'circle'
        SIZE 5
```

```
        COLOR      0 128 192
    END # Class
END # Layer

#
# Start of Layer Amostragem Sedimentos Praia
#
LAYER
    NAME "gs_praia"
    TYPE point
    CONNECTIONTYPE POSTGIS
    PROCESSING "CLOSE_CONNECTION=DEFER"
    CONNECTION "dbname=BDGM user=postgres host=localhost password=postgres port=5432"
    DATA "geom FROM gs_praia USING UNIQUE gid_gspr USING SRID=3763"
    TOLERANCE 6
    TOLERANCEUNITS pixels
    TEMPLATE void
    PROJECTION
        "init=epsg:3763"
    END
    METADATA
        "DESCRIPTION" "Sedimentos Superficiais Praia"
        "RESULT_FIELDS" "x, y, data, equipamento, stn, z_zh"
        "RESULT_HEADERS" "X, Y, Data, Equipamento, STN, Z_ZH"
        "LAYER_ENCODING" "UTF-8"
    END # Metadata
    CLASS
        # Name 'Amostragem_Vertical'
        SYMBOL 'circle'
        SIZE 5
        COLOR      58 250 226
    END # Class
END # Layer

END #Map
```

Anexo C — Scripts em linguagem de programação PHP

Sistema de autenticação

```
<body>
<form action="action.php" method="post" >
<fieldset>
<br>
  <legend>Acesso ao Portal</legend>
  <p>
    <label for="username"> </label>
    <input type="text" id="uid" name="uid" placeholder="username" required>
  </p>
  <p>
    <label for="password"> </label>
    <input type="password" id="passid" name="passid" placeholder="password"required>
  </p>
</fieldset>
  <p>
    <input type="submit" value="Entrar" />
  </p>
</form>
</body>
```

action.php

```
<?php

/* echo $_POST["uid"];*/
$db_host = 'localhost';
$db_name = 'BDGM';
$db_login = 'postgres';
$db_port = 5432;
$db_pass = 'postgres';
$username = $_POST['uid'];
$password = $_POST['passid'];

$conn_string = sprintf("host=%s port=%s dbname=%s user=%s password=%s",
    $db_host, $db_port, $db_name, $db_login, $db_pass);

$connection = pg_connect($conn_string)
    or die("Could not connect : " . pg_last_error());

$sql = "SELECT * FROM users WHERE username = '$username' AND password = '$password'";
$result = pg_query($connection, $sql);

if ($row = pg_fetch_assoc($result)) {
    header("Location: map_default.phtml");
} else {
    echo "Your username or password is incorrect!";
}
?>
```

Queries

```
<h2>Queries</h2>
<form action="querymap.phtml" target="_parent" id="choosescale" method="post" >
<fieldset>
<br>
  <legend>Cobertura geográfica</legend>
  <p>
    <label for="X1"> X1:</label>
    <input type="number" step="any" name="X1" id="X1" required><br>
  </p>
  <p>
    <label for="Y1"> Y1:</label>
    <input type="number" step="any" name="Y1" id="Y1" required><br>
  </p>
  <p>
    <label for="X2"> X2:</label>
    <input type="number" step="any" name="X2" id="X2" required><br>
  </p>
  <p>
    <label for="Y2"> Y2:</label>
    <input type="number" step="any" name="Y2" id="Y2" required><br>
  </p>
  <br>
  (X1,Y1) - coordenadas do canto inferior esquerdo <br>
  (X2,Y2) - coordenadas do canto superior direito
</fieldset>
<fieldset>
<br><br>
  <legend>Atributos</legend>
  <p>
    <label>Campanha:</label>
    <input type="text" name="campanha" id="campanha" required><br>
  </p>
  <p>
    <label>Ano:</label>
    <input type="number" name="ano" id="ano" required><br>
  </p>
</fieldset>
  <p>
    <br><br><input type="submit" value="Procurar" onclick="
    javascript:submit();$('#pmDlgContainer').hide()"/><br>
  </p>
</form>
```

querymap.html

```
$db_host = 'localhost';
$db_name = 'BDGM';
$db_login = 'postgres';
```



```
$db_port = 5432;
$db_pass = 'postgres';
$campanha = $_POST['campanha'];

$conn_string = sprintf("host=%s port=%s dbname=%s user=%s password=%s",
    $db_host, $db_port, $db_name, $db_login, $db_pass);

$connection = pg_connect($conn_string)
    or die("Could not connect : " . pg_last_error());

$sql = "SELECT gd_praia.x, gd_praia.y, gd_praia.nome, gd_praia.ambiente_sed
FROM gd_praia
WHERE gd_praia.x BETWEEN '$_POST[X1]' AND '$_POST[X2]' AND gd_praia.y BETWEEN
    '$_POST[Y1]' AND '$_POST[Y2]' AND gd_praia.nome_camp = '$campanha' AND
    gd_praia.ano = '$_POST[ano]'
UNION
SELECT gs_vertical.x, gs_vertical.y, gs_vertical.nome, gs_vertical.ambiente_sed
FROM gs_vertical
WHERE gs_vertical.x BETWEEN '$_POST[X1]' AND '$_POST[X2]' AND gs_vertical.y BETWEEN
    '$_POST[Y1]' AND '$_POST[Y2]' AND gs_vertical.nome_camp = '$campanha' AND
    gs_vertical.ano = '$_POST[ano]'
UNION
SELECT gs_plataforma.x, gs_plataforma.y, gs_plataforma.nome, gs_plataforma.ambiente_sed
FROM gs_plataforma
WHERE gs_plataforma.x BETWEEN '$_POST[X1]' AND '$_POST[X2]' AND gs_plataforma.y
    BETWEEN '$_POST[Y1]' AND '$_POST[Y2]' AND gs_plataforma.nome_camp =
    '$campanha' AND gs_plataforma.ano = '$_POST[ano]'
UNION
SELECT gd_plataforma.x, gd_plataforma.y, gd_plataforma.nome, gd_plataforma.ambiente_sed
FROM gd_plataforma
WHERE gd_plataforma.x BETWEEN '$_POST[X1]' AND '$_POST[X2]' AND gd_plataforma.y
    BETWEEN '$_POST[Y1]' AND '$_POST[Y2]' AND gd_plataforma.nome_camp =
    '$campanha' AND gd_plataforma.ano = '$_POST[ano]'
UNION
SELECT gs_praia.x, gs_praia.y, gs_praia.nome, gs_praia.ambiente_sed
FROM gs_praia
WHERE gs_praia.x BETWEEN '$_POST[X1]' AND '$_POST[X2]' AND gs_praia.y BETWEEN
    '$_POST[Y1]' AND '$_POST[Y2]' AND gs_praia.nome_camp = '$campanha' AND
    gs_praia.ano = '$_POST[ano]'
UNION
SELECT hi_sondas.x, hi_sondas.y, hi_sondas.nome, hi_sondas.ambiente_sed
FROM hi_sondas
WHERE hi_sondas.x BETWEEN '$_POST[X1]' AND '$_POST[X2]' AND hi_sondas.y BETWEEN
    '$_POST[Y1]' AND '$_POST[Y2]' AND hi_sondas.nome_camp = '$campanha' AND
    hi_sondas.ano = '$_POST[ano]'";

$result = pg_query($connection, $sql);

if (!$result) {
    echo "Problem with query " . $query . "<br/>";
    echo pg_last_error();
    exit();
}
```

```
echo("<table border=1><tr>
                                <td>x</td>
                                <td>y</td>
                                <td>nome</td>
                                <td>ambiente</td>
                                </tr>");
while ($line = pg_fetch_array($result, null, PGSQL_ASSOC)) {
echo("<tr>");
foreach ($line as $col_value => $row_value) {
    echo("<td>$row_value</td>");
}
echo("</tr>\n");
}
echo("</table>");
?>
```